Knowledge-Based Systems 201–202 (2020) 105661

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

A similarity-based two-view multiple instance learning method for classification☆

Yanshan Xiao [a], Zijian Yin [a], Bo Liu [b,*]

[a] Department of Computer Science, Guangdong University of Technology, Guangzhou, China
[b] Department of Automation, Guangdong University of Technology, Guangzhou, China

ARTICLE INFO

Article history:
Received 5 September 2019
Received in revised form 12 February 2020
Accepted 13 February 2020
Available online 15 February 2020

Keywords:
Multiple instance learning
Image classification

ABSTRACT

Multiple instance learning (MIL) has been proposed to classify the bag of instances. In practice, we may meet the problems which have more than one view data. For example, in the image classification, textual information is always used to describe the image, which can be considered as two-view data. In this paper, we propose a new similarity-based two-view multi-instance learning (STMIL) method that can incorporate two-view data into learning so as to improve classification accuracy of MIL. In order to obtain the predictive classifier, we first convert the proposed model into a convex optimization problem, and then propose a new alternative framework to solve the proposed method. We then analyze the convergence of the proposed STMIL method. The experiments have been conducted to compare the performance of our proposed method and the previous methods. The results show that our method can deliver superior performance than other methods.

1. Introduction

Multiple instance learning (MIL) [1–3] is a method evolved from supervised learning algorithm, which is proposed to address the classification of bags. MIL is originally used for drug activity prediction [1], its purpose is to build a learning system by learning the known and unknown molecules of pharmaceuticals, and to predict whether other new molecules are suitable for pharmaceuticals. In MIL, the labels in the training set are associated with sets of instances, which are called bags. In traditional MIL, if a bag contains at least one positive instance, this bag is labeled as positive; and if all the instances are negative, the bag is labeled negative. The task of MIL is to classify unlabeled bags by using the classifier into positive or negative bags. With the development of computer science and technology, MIL is applied more and more frequently in image classification [1,4,5]. In image classification, MIL treats an image as a bag, and each segmentation of an image is treated as an instance [6–8]. For example, Duan et al. [9] propose GMIL algorithm, which relaxes the constraints on the negative bag. They use *k*-means clustering algorithm to aggregate the related images into a cluster according to low-level visual features. And then, they regard a cluster as a bag, and consider each image in the bag as an instance. The image classification problem is then transformed into a MIL problem. In addition, Yingying et al. [10] propose RDMIL algorithm, they regard each image as a bag composed of different regions/patches (i.e., instances), and model image classification as a MIL problem.

The work of MIL is mainly divided into the following three categories [11]. In the first category [12], MIL trains the classifier by setting the instance tags in the bag as positive, and uses the standard supervised learning methods or iterative frameworks to build the classifier. Paul et al. [13] propose the MILBoost, which uses the label of the bag to initialize the label of instances, so that the label of the instance is the same as the label of the bag, and then uses a boosting framework to learn the classifier. In the second category [14], MIL establishes a mechanism that maps instances to "bag-level" training vectors. For example, Jia et al. [15] propose the MILDM, which uses instances to map each bag to a new feature space to get the best instance of the bag. MILDM maps the original feature space's bag into the new feature space by discriminative instance pool, and the training classifier is used to predict the class label. In the third category [16], MIL focuses on selecting a subset of instances from the positive bag to learn the classifier. For an example, Liming et al. [17] propose the MILMPC, which is a method for instance selection based on the MIL framework. MILMPC uses the multi-point concept to deal with the problem of instance selection, that is, each possible concept is associated with a similar set of instances. The candidate multi-point concept is derived from the concept extraction of the instance in the positive bag. The method then calculates the

☆ No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.knosys.2020.105661.

* Corresponding author.
E-mail addresses: xiaoyanshan@189.cn (Y. Xiao), 229715148@qq.com (Z. Yin), csbliu@aliyun.com, csboliu@163.com (B. Liu).

https://doi.org/10.1016/j.knosys.2020.105661
0950-7051/

Fig. 1. Web images with text descriptions.

relevance of each candidate concept to the positive class, and adds the concept with the highest relevance to a concept set until no new candidate concept is added.

Although there is a lot of research on MIL, however, we may meet the multi-instance data which consists of more than one view in practice. For example, in the image classification, in addition to image information, there are some text information in the image classification [18]. These text information can be applied to image classification to improve classification accuracy [19,20]. For example, on the Internet, when a user shares an image, different users can comment on the image. As shown in Fig. 1, the dog is on the left, and the cat is on the right. The text below these images are the comments and descriptions of these images by different users A, B and C. Through the above example, we can find that these comments or descriptions may be valuable for images retrieval and classification. This kind of problem is also named as image with annotation problem [21–24]. Although deep learning has been used in MIL for image with annotation problem [21–24]; however, most of them use the existing network for MIL, and it is hard to design network architecture especially for MIL.

In this paper, we propose a new similarity-based two-view MIL (STMIL), based on our previous developed SMILE method [25] for single task learning, STMIL method can incorporate both two-view data into the MIL. The proposed method can be used in the problem of image classification associated with a number of description text. We first propose similarity-based two-view MIL model, and then develop a new multiple instance framework to obtain the predictive classifier based on the proposed model. The main contributions of our work are as follows:

(1) We propose a new similarity-based two-view method (STMIL), based on the basic MIL method [25] to incorporate two-view data in to a learning model, such that we can deliver a more accurate predictive classifier. In the STMIL model, we put forward to utilize the hyperplane constrains of two views to maintain a certain harmonious relationship between the hyperplanes in the two views, which can be embedded in the similarity-based two-view model. In addition, we utilize Lagrange method to convert the similarity-based multiple instance model into its Dual form such that we can solve the objective model to obtain the initial SVM classifier.
(2) In order to obtain the multiple instance classifier, we develop a new alternative framework to solve the similarity-based two view MIL, and obtain the predictive classifier. In this procedure, we first initialize the positive example for each bag at its own view, and then update the multiple instance classifier synchronously. The proposed method is then utilized for image classification with description text. We further analyze the convergence of the proposed STMIL method.
(3) The extensive experiments have been conducted to evaluate the performance of our proposed STMIL method. We evaluate our proposed method on the NUS-WIDE and Flickr30k Entities datasets. The results show that our method demonstrates highly competitive classification accuracy and shows less sensitivity to the labeling noise than the existing MIL methods.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 proposes our similarity-based two-view MIL. Experiments are conducted in Section 4. The conclusion is presented in Section 5.

2. Related work

In this section, we will review the previous works related to our proposed similarity-based two-view MIL method. In Section 2.1, we review the multiple view learning, and in Section 2.2, we review the previous work on MIL.

2.1. Multiple views learning

The initial multi-view learning algorithm is presented in the work [26], here "views" means the data coming from multiple sources or different feature subsets. In some practical problems, the same object can be always described in a variety of ways or angles, and each view data can describe the object from its own perspective. For example in the video classification, the image information and voice information can be two different features, and they can be regarded as two-view data. In addition, for the image classification with textual description information, the image and the text are always regarded as two-view data, which can be utilized in the following classification procedure. By training the classifier with multi-view learning, the performance of the classifier in unbalanced classification can be improved [27]. For examples, Zhe et al. [27] propose a learning framework consisting of fisher kernel and Bi-Bagging method to solve the problem of imbalance. Xijiong et al. [28] propose the regularized multi-view least squares twin support vector machines to generate binary classifiers and improve the generalization performance of multiple different feature sets. There are many algorithms in the development of multiple views learning, such as co-training [29], multi-kernel learning [30–33], subspace learning algorithm [34, 35].

The co-training algorithm is proposed to solve the problem of semi-supervised multiple views learning. In this method, the classifier is first trained by marking data on the two views space, and each classifier selects a number of data with higher predictive confidence in the unlabeled data and then adds the labeled data to another related classifier. Data centralization enables the other party to update these new tag data and iterates the process until a certain stop condition is reached. Meng et al. [36] propose a weighted co-training algorithm for cross-domain image sentiment classification to predict the emotional polarity of images. They train the two sentiment classifiers with images and corresponding text annotations, and set the similarity between the source and target domains. Co-training algorithm can also be used for hyperspectral image classification. Xiangrong et al. [37] propose a semi-supervised method based on a modified co-training process with spectral and spatial views. They use the original spectral features and the 2-D Gabor features extracted from the spatial domain as two different views of the common training. Their approach effectively improves the performance of co-training.

Multiple kernel learning (MKL) combined with support vector machine is used to solve the problem of non-linear classification of training samples. In a large number of kernel functions, the commonly used Gaussian kernel function [38,39] has considerable flexibility and maps data low-dimensional space to high-dimensional space. Multilinear subspace learning is a dimensionality reduction method by directly mapping high-dimensional tensor data to low-dimensional space. When the data is distributed in a high-dimensional space, we do not deal with it very effectively, because the complexity of the classification model is correspondingly increased due to the too high dimension, which ultimately leads to the classification model being easy to overfit. Xiangping et al. [40] propose a multiple feature fusion method based on multi-class multiple kernel learning. Their method fuses many features, which can effectively avoid decomposing multiple types of problems into multiple binary classifications and directly get the classifier. Qi et al. [41] propose a method based on sparse coding and multiple kernel learning. They add the spatial information by dividing the image with the spatial pyramid, and use nonlinear SVM for image classification, and achieve optimal trade-offs between different kernels.

Multiline subspace learning can solve this problem very effectively. There are two principles of multiple views learning, consensus principle and complementarity principle. In each view, there is a consensus between them, that is, common constraints and they must work together to fully describe the data. The single view is a one-sided description of the data. Jun et al. [42] propose semi-supervised multi-modal subspace learning (SS-MMSL). They use the data distribution revealed by unlabeled data to enhance subspace learning, and use alternating iterative optimization algorithms to explore the complementary features of different modes. Xiaozhao et al. [43] propose an image classification method called robust latent subspace learning (RLSL). They translate the RLSL problem into a joint optimization problem for potential subspace learning and classification model parameter prediction. RLSL combines feature learning with classification, making learning data representation in potential subspaces more discriminating. Xiao et al. [44] propose MRSLA which utilizes multi-view data sources to discover potential disease-associated miRNAs. The MRSLA method projects the miRNA-disease associations into two subspaces, and uses a low-rank approximation-based recommendation method to predict disease miRNA candidates. Xin et al. [45] propose a novel multi-view discriminant analysis based on Hilbert–Schmidt Independence Criterion (HSIC) and canonical correlation analysis (CCA). They use HSIC to identify lower dimensional distinguishable common subspaces, and use CCA to achieve maximum correlation between different views in the common subspace.

In addition, multiple views learning can also be applied to clustering of data, known as multi-view clustering [46,47]. Yiling et al. [46] propose a multi-task multi-view clustering algorithm in heterogeneous situations based on Locally Linear Embedding (LLE) and Laplacian Eigenmaps (LE) methods (L3E-M2VC). This method solves the problem of insufficient information about the label set and different label sets in all learning task. This method maps samples in multiple views to a common view, transforms the common view into a discriminative task space, and clusters the data by k-means method. Hao et al. [47] propose a general Graph-Based System (GBS) for multi-view clustering. The graph matrix of the view data is obtained by extracting the feature matrix of the view, and then the clustering is obtained by fusing the graph matrix.

The two-view learning is a form of multi-view learning with arbitrary number of views. Kernel canonical correlation analysis (KCCA) is an effective preprocessing step that can improve the performance of SVM when two views of the same phenomenon are available. According to this conclusion, Jason et al. [48] propose the SVM-2K. Considering the existence of labeled and unlabeled data, such as spam detection, Guangxia et al. [49] propose the two-view transductive method. The method constructs two views with labeled and unlabeled data to train classifier, and applies global constraints to each labeled and unlabeled data. Canonical correlation analysis (CCA) requires that the data of two views must be matched, and canonical principal angles correlation analysis (CPACA) [50] makes the classic CCA out of this limit. The basic idea is that the correlation of two views is represented by the similarity between two subspace spanned by the principal components. Two-view learning can also be applied to emotion recognition system [51], which is used to classify facial expression in video. The basic idea is to extract two basic facial expression features and construct two-view classifiers.

2.2. Multiple instance learning

The initial MIL algorithms are presented in [52–56], which defines "bags" as a set of multiple instances. If all the sample markers are known, it is a supervised learning problem. MIL has wide applications and solves learning problems with ambiguity on training samples. However, when there are many samples of labels we do not know, MIL is useful and its solution to the problem is iterative optimization. Assuming that all the markers are known (all the instances in the positive bags are labeled as positive labels, the instances in the negative bags are negative labels). Then, a classification model can be obtained by some supervised learning methods, through which we can predict each testing sample with the model [57–59].

Maron et al. [53] propose a framework which is called Diverse Density (DD) to solve multiple-instance problems. The DD is a measure of the degree of integration between positive bags and negative bags. They can find the intersection point (the required concept) and a set of feature weights that lead to optimal crossover by maximizing the density. EM-DD [55] selects an instance that is most consistent with the current assumptions in each positive bag, so as to predict the unknown bag. EM-DD repeatedly guesses subset of instances from positive bags to learn classifiers. Andrews et al. [52] propose MI–SVM and MI–SVM to solve MI learning problems. MI–SVM maximizes the bag margin by utilizing an iterative method to learn the SVM classifier, while MI–SVM maximizes the instance margin through the possible label allocation and hyperplane. Correia et al. [59] propose θ-MIL to detect polarity of movie reviews, they use the IMDb movie review corpus dataset to test θ-MIL for SVM and achieve good results. They improve the MI–SVM and MI–SVM algorithm and obtain the θ-MI–SVM and θ-MI–SVM algorithms. Lixin et al. [9] propose the generalized MIL (GMIL) to improve web image search. Their GMIL relaxes the constraint on negative bags on the traditional MIL, they allow positive instances with a certain proportion in negative bags, and the constraints in positive bags are the same as those of the traditional ones. There may be noise in the data, robust model fitting methods [60,61] can solve this problem. Robust model fitting methods can effectively segment multi-structure data, even though these data are heavily contaminated by outliers. They first use a greedy search strategy to sample the dataset to generate the model hypotheses, then use different detectors to detect whether the parameters of the generated hypotheses are correct.

In addition, deep multiple instance learning (DMIL) has been studied. The DMIL framework [21] considers a set of tags for an image as a "bag", and considers one of tag in a set as an instance. It then denoises the images and keywords using CNN and DNN, respectively. An end-to-end learning framework based on DMIL [22] can be used to classify multispectral (MS) and

panchromatic (PAN) images. The framework uses two instances, one for capturing spatial information of the PAN and the other for describing the spectral information of the MS. Simple fusion features are produced by direct connection of features obtained from these two instances. By incorporating simple fusion features into the fusion network process, high-level fusion features are obtained and a classifier is obtained. Depend on the architecture of the network, DMIL embeds MIL into the neural networks [62, 63]. Therefore, it is hard to design neural network especially for MIL [16,64].

In this paper, we propose similarity-based two-view MIL (STMIL) algorithm based on our previous developed SMILE method to solve the problem of image classification with text information. An image is regarded as a bag and all text of the image is also regarded as a bag; however, label of instance is still ambiguous. And then, we propose two-view multiple instance classifier for image classification.

3. Similarity-based two-view multiple instance learning

In this section, we will propose a new similarity-based two-view MIL method and apply it to the image classification with description text. In Section 3.1, we will discuss the similarity model and two-view data of MIL. In Section 3.2, we present the proposed STMIL-SVM method, and give the solution process and results. In Section 3.3, we propose a new alternative framework for STMIL method and analyze its convergence. We present the decision boundary determination of the method.

3.1. Data model and two-view data generation

We introduce the proposed similarity-based data model to describe data as follows. An instance $\boldsymbol{x_i}$ from a multiple instance bag is denoted as $\{\boldsymbol{x_i}, m^+(\boldsymbol{x_i}), m^-(\boldsymbol{x_i})\}$, where $m^+(\boldsymbol{x_i})$ and $m^-(\boldsymbol{x_i})$ represent the similarity of $\boldsymbol{x_i}$ to the positive and the negative class, respectively, and it has $0 \le m^+(\boldsymbol{x_i}) \le 1$ and $0 \le m^-(\boldsymbol{x_i}) \le 1$. Let $m^+(\boldsymbol{x_i}) = 1$ and $m^-(\boldsymbol{x_i}) = 0$ if the label of the instance $\boldsymbol{x_i}$ is positive. And let $m^+(\boldsymbol{x_i}) = 0$ and $m^-(\boldsymbol{x_i}) = 1$ if the label of the instance $\boldsymbol{x_i}$ is negative. Hence, there are three possibilities: $\{\boldsymbol{x_i}, 1, 0\}$ means that $\boldsymbol{x_i}$ is a positive instance, $\{\boldsymbol{x_i}, 0, 1\}$ means that $\boldsymbol{x_i}$ is a negative instance, and $\{\boldsymbol{x_i}, m^+(\boldsymbol{x_i}), m^-(\boldsymbol{x_i})\}$ means that $\boldsymbol{x_i}$ is an ambiguous instance, and $0 < m^+(\boldsymbol{x_i}) < 1$ and $0 < m^-(\boldsymbol{x_i}) < 1$.

For A-view, we let S_p^{A+} store positive candidates in the positive bags, S_a^{A+} store the remaining instances except for the positive candidates in the positive bags, and S_n^{A-} store the instances from the negative bags, respectively. For instances in S_a^{A+}, they have bi-memberships $\{m^+(\boldsymbol{x_i}), m^-(\boldsymbol{x_i})\}$ towards the positive and negative classes respectively.

Definition 1 (*Set-Based Similarity*)**.**: Given an instance $\boldsymbol{x}$ and a subset S, the similarity of $\boldsymbol{x}$ to S is defined as: [19]

$$R(\boldsymbol{x}, S) = \frac{1}{2}\sum_{\boldsymbol{x_i}\in S} e^{-\|\varphi(\boldsymbol{x})-\varphi(\boldsymbol{x_i})\|} \tag{1}$$

Where $\varphi(\cdot)$ is a nonlinear mapping function that is used to map the instance $\boldsymbol{x}$ or $\boldsymbol{x_i}$ into the feature space. Then, both memberships are calculated as follows:

$$m_A^+(\boldsymbol{x_i}) = \frac{1}{2}[R(\boldsymbol{x}, S_p^{A+}) + 1 - R(\boldsymbol{x}, S_n^{A-})] \tag{2}$$

$$m_A^-(\boldsymbol{x_i}) = \frac{1}{2}[R(\boldsymbol{x}, S_n^{A-}) + 1 - R(\boldsymbol{x}, S_p^{A+})] \tag{3}$$

Based on the above definitions, we have the same explanation for B-view. If an instance is close to the positive candidates while far from the negative instances, it has larger membership towards the positive while lower membership towards the negative. For A-view, we further let $S^{A'} = S_p^{A+} \cup S_a^{A+}$ and $S^{A''} = S_a^{A+} \cup S_n^{A-}$. Similarly, we also let $S^{B'} = S_p^{B+} \cup S_a^{B+}$ and $S^{B''} = S_a^{B+} \cup S_n^{B-}$ for B-view.

For the problem of image classification with description text, we convert the image and text into multiple instance data form using the existing methods and extract the features from the image data as follows. The scale-invariant feature transform (SIFT) [65] feature is based on the point of interest of some local appearance regardless of the size and rotation of the image. The tolerance for changes in light, noise, and micro-angle of view is also quite high. Based on these characteristics, they are highly significant and relatively easy to extract. We then use SIFT algorithm to extract feature points of the image, and use different clustering methods, (i.e., k-means [66], EM clustering [67] and DBSCAN [68]) to cluster all the extracted feature points, then construct a bag of word (BOW) descriptor for each image. We can also use the segmentation algorithms, (i.e., GrabCut [69] and MILCut [70]) to divide the image into the A-view. For text feature extraction, we use the term frequency–inverse document frequency (TF–IDF) [71] method, which is formed into B-view. The main idea of TF–IDF is that if a word or phrase appears in an article with a high frequency and rarely appears in other articles, the word or phrase is considered to have good class distinguishing ability and is suitable for classification. For details, see the experiment section.

In this paper, we let $(B_I^A, Y_I^A), I = 1, 2, \ldots |Y^A|$ and $(B_J^B, Y_J^B), J = 1, 2, \ldots |Y^B|$ denote the training sets for the A-view and B-view. Here, $|Y^A|$ and $|Y^B|$ are the number of bags for the A-view and B-view respectively. For $(B_I^A, Y_I^A), I = 1, 2, \ldots |Y^A|$, where B_I denotes a bag which contains a number of instances, we utilize $\boldsymbol{x_i}$ (where $\boldsymbol{x_i} \in B_I$) to denote an instance of B_I; Y_I denotes the label of the bag: if $Y_I = 1$, then at least one positive instance exists in B_I, if $Y_I = -1$, then each instance in B_I is negative. For $(B_J^B, Y_J^B), J = 1, 2, \ldots |Y^B|$, we have the same explanation. In the traditional MIL, there is at least one positive instance in a positive bags, all negative instances in a negative bag. For $(\boldsymbol{x_i}, y_i)$, where y_i is the label of an instance $\boldsymbol{x_i}$. The constraints are as follows:

$$\begin{aligned} &\sum_{i:\boldsymbol{x_i}\in B_I} \frac{y_i+1}{2} \ge 1, \quad \text{for} \quad Y_I = 1 \\ &\sum_{i:\boldsymbol{x_i}\in B_I} \frac{y_i+1}{2} \le 0, \quad \text{for} \quad Y_I = -1 \end{aligned} \tag{4}$$

In MIL, even if we know that positive bags contain positive instances, we still do not know the real labels of the instances in positive bags. In MIL model, we suppose that $D = \{(B_1^+, Y_1^+), \ldots, (B_{N^+}^+, Y_{N^+}^+), (B_1^-, Y_1^-), \ldots, (B_{N^-}^-, Y_{N^-}^-)\}$ denotes a set of training bags, where B_i^+ represents a positive bag with a positive label $Y_i^+ = +1$; B_i^- represents a negative bag with a negative label $Y_i^- = -1$; where N^+ and N^- are the numbers of positive bags and negative bags, respectively. Each bag contains a set of instances, we use $\boldsymbol{x}$ to represent an instance, and $y = \pm 1$ denotes label of an instance. We then convert image with text into multi-instance data.

3.2. STMIL method

In our proposed approach, suppose we train SVM on (B_I^A, Y_I^A) for the A-view and on (B_J^B, Y_J^B) for the B-view. We assume that $f_{A/B} = \omega_{A/B}\phi(\boldsymbol{x}) + b_{A/B}$ are two hyperplanes for both views. We propose STMIL-SVM method based on our previous work on MIL [25]. This leads to the following minimization problem of

STMIL-SVM and the formulation is given by:

$$\begin{aligned}\min_{\omega,b,\xi}\quad & \frac{1}{2}\|\omega_A\|^2+\frac{1}{2}\|\omega_B\|^2+D_A\{\sum_{S^{A'}}m_A^+[\phi(\boldsymbol{x_i})]\xi_i+\sum_{S^{A''}}m_A^-[\phi(\boldsymbol{x_j})]\xi_j\}\\ &+D_B\{\sum_{S^{B'}}m_B^+[\phi(\boldsymbol{x_k})]\xi_k+\sum_{S^{B''}}m_B^-[\phi(\boldsymbol{x_h})]\xi_h\}\\ &+C\sum_{i=1}^{n}\eta_i\\ s.t.\quad & |\omega_A\phi(\boldsymbol{x_i})+b_A-\omega_B\phi(\boldsymbol{x_j})-b_B|\le\sum_{i=1}^{n}\eta_i+\varepsilon\\ & \omega_A\phi(\boldsymbol{x_i})+b_A\ge 1-\xi_i,\quad \omega_A\phi(\boldsymbol{x_j})+b_A\le -1+\xi_j\\ & \omega_B\phi(\boldsymbol{x_k})+b_B\ge 1-\xi_k,\quad \omega_B\phi(\boldsymbol{x_h})+b_B\le -1+\xi_h\\ & \xi_i\ge 0,\quad \xi_j\ge 0,\quad \xi_k\ge 0,\quad \xi_h\ge 0,\quad \eta_i\ge 0\end{aligned}\tag{5}$$

where D_A and D_B control the preference of two views. If $D_A > D_B$, A-view is preferred to B-view; otherwise, B-view is preferred to A-view. Parameter C is a parameter to balance the margin and errors. And ξ_i, ξ_j, ξ_k and ξ_h are slack variables. In A-view, for instance $\boldsymbol{x_i}$ in S_p^{A+}, its membership is $m_A^+(\boldsymbol{x_i})=1$ towards positive class. Then, each instance in $S^{A'}$ has $m_A^+(\boldsymbol{x_i})$ towards the positive class. We set $m_A^-(\boldsymbol{x_j})=-1$ for each instance in S_n^{A-}, thus each instance in $S^{A''}$ has $m_A^-(\boldsymbol{x_j})$ towards the negative class. Similarly, we have $m_B^+(\boldsymbol{x_k}), m_B^-(\boldsymbol{x_h})$ for B-view. For constraint condition $|f_A-f_B|\le\sum_{i=1}^{n}\eta_i+\varepsilon$, which denotes the constraint of two views. f_A and f_B denotes the similarity-based SVM decision functions of A-view and B-view, respectively. η_i is a variable, which imposes consensus between the two views, and ε is a slack variable for allowing some instances to violate the constraint. To maintain a certain harmonious relationship between the hyperplanes in the two views, that is, the values between each pair of samples in different views are not much different. In addition, $\phi(\cdot)$ is mapping function, which maps the data from input space into a feature space, and the inner product of two vectors in feature space can be calculated by a kernel function, that is $K(\boldsymbol{v},\boldsymbol{x})=\phi(\boldsymbol{v})\cdot\phi(\boldsymbol{x})$.

Problem Solution

We solve the optimization problem in (5) by introducing the Lagrangian [19] method, and can get Theorem 1.

Theorem 1. *The optimization problem in (5) can be resolved by the optimization problem (6): We introducing Lagrange multipliers $\alpha_i, \alpha_j, \alpha_k$, and α_h for the instances in $S^{A'}, S^{A''}, S^{B'}$ and $S^{B''}$. Then we can arrive the solution of problem (6) is to resolve the dual problem:*

$$\begin{aligned}\max F=&\sum_{S^{A'}}\alpha_i+\sum_{S^{A''}}\alpha_j+\sum_{S^{B'}}\alpha_k+\sum_{S^{B''}}\alpha_h-(\beta_i+\beta_j)\varepsilon\\ &+\alpha_k\alpha_hK(\boldsymbol{x_k},\boldsymbol{x_h})+\alpha_j(\alpha_i+\beta_i-\beta_j)K(\boldsymbol{x_i},\boldsymbol{x_j})\\ &+\alpha_k(\beta_i-\beta_j)K(\boldsymbol{x_j},\boldsymbol{x_k})-\alpha_h(\beta_i-\beta_j)K(\boldsymbol{x_h},\boldsymbol{x_j})\\ &-\frac{1}{2}(\alpha_i+\beta_i-\beta_j)^2K(\boldsymbol{x_i},\boldsymbol{x_i})-\frac{1}{2}\alpha_j^2K(\boldsymbol{x_j},\boldsymbol{x_j})\\ &-\frac{1}{2}\alpha_h^2K(\boldsymbol{x_h},\boldsymbol{x_h})-\frac{1}{2}\alpha_k^2K(\boldsymbol{x_k},\boldsymbol{x_k})-\frac{1}{2}(\beta_i-\beta_j)K(\boldsymbol{x_j},\boldsymbol{x_j})\\ s.t.\quad & 0\le\alpha_i\le D_Am_A^+[\phi(\boldsymbol{x_i})],\quad 0\le\alpha_j\le D_Am_A^-[\phi(\boldsymbol{x_j})]\\ & 0\le\alpha_k\le D_Bm_B^+[\phi(\boldsymbol{x_k})],\quad 0\le\alpha_h\le D_Bm_B^-[\phi(\boldsymbol{x_h})]\\ & 0\le\gamma_i\le D_Am_A^+[\phi(\boldsymbol{x_i})],\quad 0\le\gamma_j\le D_Am_A^-[\phi(\boldsymbol{x_j})]\\ & 0\le\gamma_k\le D_Bm_B^+[\phi(\boldsymbol{x_k})],\quad 0\le\gamma_h\le D_Bm_B^-[\phi(\boldsymbol{x_h})]\\ & 0\le\beta_i\le C,\quad 0\le\beta_j\le C,\quad 0\le\delta_i\le C\end{aligned}\tag{6}$$

Proof. The optimization problem in (5) can be converted to the dual form by differentiating the Lagrangian function with the original variables $\omega_A, \omega_B, \xi_i, \xi_j, \xi_k$, and ξ_h. We introduce the Lagrange multipliers $\alpha_i\ge 0, \alpha_j\ge 0, \alpha_k\ge 0, \alpha_h\ge 0, \beta_i\ge 0, \beta_j\ge 0, \gamma_i\ge 0, \gamma_j\ge 0, \gamma_k\ge 0, \gamma_h\ge 0$ and $\delta_i\ge 0$. Based on the defined Lagrange multipliers, the Lagrangian function of the objective function in (5) can be given as:

$$\begin{aligned}L=&\frac{1}{2}\|\omega_A\|^2+\frac{1}{2}\|\omega_B\|^2+C\sum_{i=1}^{n}\eta_i-\sum_{i=1}^{n}\delta_i\eta_i\\ &+D_A\{\sum_{S^{A'}}m_A^+[\phi(\boldsymbol{x_i})]\xi_i+\sum_{S^{A''}}m_A^-[\phi(\boldsymbol{x_j})]\xi_j\}\\ &+D_B\{\sum_{S^{B'}}m_B^+[\phi(\boldsymbol{x_k})]\xi_k+\sum_{S^{B''}}m_B^-[\phi(\boldsymbol{x_h})]\xi_h\}\\ &-\sum_{S^{A'}}\alpha_i[\omega_A\phi(\boldsymbol{x_i})+b_A-1+\xi_i]-\sum_{i=1}^{n}\gamma_i\xi_i\\ &+\sum_{S^{A''}}\alpha_j[\omega_A\phi(\boldsymbol{x_j})+b_A+1-\xi_j]-\sum_{j=1}^{n}\gamma_j\xi_j\\ &-\sum_{S^{B'}}\alpha_k[\omega_B\phi(\boldsymbol{x_k})+b_B-1+\xi_k]-\sum_{k=1}^{n}\gamma_k\xi_k\\ &+\sum_{S^{B''}}\alpha_h[\omega_B\phi(\boldsymbol{x_h})+b_B+1-\xi_h]-\sum_{h=1}^{n}\gamma_h\xi_h\\ &-\sum_{i=1}^{n}\beta_i[\omega_A\phi(\boldsymbol{x_i})+b_A-\omega_B\phi(\boldsymbol{x_j})-b_B+\sum_{i=1}^{n}\eta_i+\varepsilon]\\ &+\sum_{j=1}^{n}\beta_j[\omega_A\phi(\boldsymbol{x_i})+b_A-\omega_B\phi(\boldsymbol{x_j})-b_B-\sum_{i=1}^{n}\eta_i-\varepsilon]\end{aligned}\tag{7}$$

Differentiating the Lagrangian function (7) with $\omega_A, \omega_B, b_A, b_B, \xi_i, \xi_j, \xi_k, \xi_h$ and η_i, the following equations are obtained:

$$\frac{\partial L}{\partial\omega_A}=\omega_A+(-\alpha_i-\beta_i+\beta_j)\phi(\boldsymbol{x_i})+\alpha_j\phi(\boldsymbol{x_j})=0\tag{8}$$

$$\frac{\partial L}{\partial\omega_B}=\omega_B-\alpha_k\phi(\boldsymbol{x_k})+\alpha_h\phi(\boldsymbol{x_h})+(\beta_i-\beta_j)\phi(\boldsymbol{x_j})=0\tag{9}$$

$$\frac{\partial L}{\partial b_A}=-\alpha_i+\alpha_j-\beta_i+\beta_j=0\tag{10}$$

$$\frac{\partial L}{\partial b_B}=-\alpha_k+\alpha_h+\beta_i-\beta_j=0\tag{11}$$

$$\frac{\partial L}{\partial\xi_i}=D_Am_A^+[\phi(\boldsymbol{x_i})]-\alpha_i-\gamma_i=0\tag{12}$$

$$\frac{\partial L}{\partial\xi_j}=D_Am_A^-[\phi(\boldsymbol{x_j})]-\alpha_j-\gamma_j=0\tag{13}$$

$$\frac{\partial L}{\partial\xi_k}=D_Bm_B^+[\phi(\boldsymbol{x_k})]-\alpha_k-\gamma_k=0\tag{14}$$

$$\frac{\partial L}{\partial\xi_h}=D_Bm_B^-[\phi(\boldsymbol{x_h})]-\alpha_h-\gamma_h=0\tag{15}$$

$$\frac{\partial L}{\partial\eta_i}=C-\beta_i-\beta_j-\delta_i=0\tag{16}$$

We can get $\omega_{A/B}$ according to (8) and (9) as follows:

$$\omega_A=(\alpha_i+\beta_i-\beta_j)\phi(\boldsymbol{x_i})-\alpha_j\phi(\boldsymbol{x_j})\tag{17}$$

$$\omega_B=\alpha_k\phi(\boldsymbol{x_k})-\alpha_h\phi(\boldsymbol{x_h})+(\beta_j-\beta_i)\phi(\boldsymbol{x_j})\tag{18}$$

From the Kuhn–Tucker Theorem, we substitute (10)–(18) into the Lagrangian function (7). Next, we get the (19) below.

$$L = \sum_{S^{A'}} \alpha_i + \sum_{S^{A''}} \alpha_j + \sum_{S^{B'}} \alpha_k + \sum_{S^{B''}} \alpha_h - (\sum_{i=1}^{n} \beta_i + \sum_{j=1}^{n} \beta_j)\varepsilon - \frac{1}{2}\|\omega_A\|^2 - \frac{1}{2}\|\omega_B\|^2 \tag{19}$$

We further substitute (17) and (18) into (19), the Wolfe dual of (5) can be obtained (6). For the constraints in (6), $0 \le \alpha_i \le D_A m_A^+(\boldsymbol{x_i})$ and $0 \le \gamma_i \le D_A m_A^+(\boldsymbol{x_i})$ can be obtained, since $\alpha_i \ge 0$ and $\gamma_i \ge 0$ in (12). Similarly, $0 \le \alpha_j \le D_A m_A^-(\boldsymbol{x_j})$ and $0 \le \gamma_j \le D_A m_A^-(\boldsymbol{x_j})$ can be obtained, because $\alpha_j \ge 0$ and $\gamma_j \ge 0$ in (13); $0 \le \alpha_k \le D_B m_B^+(\boldsymbol{x_k})$ and $0 \le \gamma_k \le D_B m_B^+(\boldsymbol{x_k})$ can be obtained because of $\alpha_k \ge 0$ and $\gamma_k \ge 0$ in (14); $0 \le \alpha_h \le D_B m_B^-(\boldsymbol{x_h})$ and $0 \le \gamma_h \le D_B m_B^-(\boldsymbol{x_h})$ can be obtained due to $\alpha_h \ge 0$ and $\gamma_h \ge 0$ in (15). Finally, $0 \le \beta_i \le C$, $0 \le \beta_j \le C$ and $0 \le \delta_i \le C$ can be obtained according to $\beta_i \ge 0$, $\beta_j \ge 0$ and $\delta_i \ge 0$ in (16).

3.3. Alternative framework for STMIL method

In order to solve the problem of two-view MIL, we propose a new framework for STMIL method as follows. The STMIL approach is presented in Algorithm 1. After obtain the similarity-based two-view model, we propose alternative framework on top of SMILE and MI–SVM [72] methods to solve the problem of two-view MIL. We first initialize the positive example for each bag for each view data, and then update the multiple instance classifier for each view at the same time, and the target is to train two classifiers. For Algorithm 1, we let S_p^{A+}, S_p^{B+} store positive candidates in the positive bags in A-view and B-view, S_a^{A+}, S_a^{B+} store the remaining instances except for the positive candidates in the positive bags in A-view and B-view respectively. And let S_n^{A-}, S_n^{B-} store the instances from the negative bags in A-view and B-view, respectively. Firstly, we construct bags by utilizing k-means algorithm to aggregate all the instances according to their visual feature and textual feature. Secondly, we initialize label of all bags according to (2) and (3). Thirdly, $\alpha^A, \alpha^B, \beta, S_p^{A+}, S_p^{B+}, S_a^{A+}, S_a^{B+}, S_n^{A-}, S_n^{B-}$ are initialized, and two positive candidates $\boldsymbol{x_i^A}, \boldsymbol{x_i^B}$ are randomly determined in A. Fourthly, S_p^{A+}, S_p^{B+} are updated by replacing the positive candidate in the tth iteration, i.e., $\boldsymbol{x_{k^{t-1}}^A}, \boldsymbol{x_{k^{t-1}}^B}$ with $\boldsymbol{x_{k^t}^A}, \boldsymbol{x_{k^t}^B}$. Fifthly, we arrive A-view and B-view by separating the text from the images. Lastly, we solve the optimization problem (6) to calculate ω and b.

For the convergence of the method, because of the value of F is nonnegative and decreases monotonically, Algorithm 1 can converge after a finite number of steps. The value of $F(\cdot)$ is determined by the Lagrange multipliers $\alpha^{A/B}, \beta^{+/-}, \gamma^{A/B}$ and δ and the positive candidates in subset S_p^{A+} and S_p^{B+}. We alternatively optimize Lagrange multipliers and positive candidate to maximize the values $F(\cdot)$. Based on this, we have the following relations:

$$F^{(t)} \ge F^{(t-1)} \tag{20}$$

It is seen that the value of $F(\cdot)$ is monotonically increased during the whole process of optimization. Therefore, the procedure will converge until $|F^{(t-1)} - F^{(t)}| \le \epsilon F^{(t-1)}$ satisfies. Here ϵ is a threshold, which is set to be 0.01 in the experiments.

After solving the dual form (5), $\omega_{A/B}$ and $b_{A/B}$ in A/B-view are obtained. The decision function to predict the instance and bag label is given by

$$y_{A/B}(\boldsymbol{x}) = \begin{cases} +1, & \omega_{A/B}\varphi(\boldsymbol{x}) + b_{A/B} \ge 0 \\ -1, & \omega_{A/B}\varphi(\boldsymbol{x}) + b_{A/B} < 0 \end{cases} \tag{21}$$

where $y_{A/B}(\boldsymbol{x})$ denotes the label of $\boldsymbol{x}$.

Algorithm 1 STMIL-SVM

Require: Training bags $\{(B_I, Y_I)|I = 1, ..., N\}$ in two views.

```
Constructing bags by using k-means algorithm to aggregate all the instances;
Initialize label Y_I of all bags B_I according to (2) and (3);
Produce two views by separating the text from the images;
Initialize α^A, α^B, β, S_p^{A+}, S_p^{B+}, S_a^{A+}, S_a^{B+}, S_n^{A−}, S_n^{B−};
Let t = 0;
repeat
t = t + 1;
for (each positive bag B_A^+, B_B^+ in A-view, B-view) do
    for (each instance x_i^A, x_i^B in A-view, B-view) do
        Let x_i^A, x_i^B be the positive candidate of B_A^+, B_B^+;
        S_p^{A+'} ← S_p^{A+} and S_p^{B+'} ← S_p^{B+};
        Update S_p^{A+'}, S_p^{B+'} by replacing x_{k^{t-1}}^A, x_{k^{t-1}}^B
        with x_i^A, x_i^B;
        S_a^{A+} ← D − S_p^{A+'} − S_n^{A−};
        S_a^{B+} ← D − S_p^{B+'} − S_n^{B−};
        Calculate the value of f^A, f^B denoted as F(x_i);
    end for
    Update S_p^{A+}, S_p^{B+} by replacing x_{k^{t-1}}^A, x_{k^{t-1}}^B with x_{k^t}^A, x_{k^t}^B;
    Obtain new positive candidate returns arg max F(x_i);
end for
S_a^{A+} ← D − S_p^{A+} − S_n^{A−}, S_a^{B+} ← D − S_p^{B+} − S_n^{B−};
Compute m^+(x_i), m^−(x_i) according to (2) and (3);
Obtain α^A, α^B, b_A, b_B and F by solving QP in (6) based on
S_p^{A+}, S_p^{B+}, S_a^{A+}, S_a^{B+}, S_n^{A−}, S_n^{B−};
α_{(t)}^A ← α^A, α_{(t)}^B ← α^B, F^{(t)} ← F;
until |F^{(t-1)} − F^{(t)}| ≤ εF^{(t-1)};
Output(ω_A, b_A, ω_B, b_B);
```

The objective of MIL is to train two classifiers on the bag data and utilize the obtained classifiers to predict the labels of bags. Based on the instance-level decision function (21), the decision function to predict the bag label is given as follows:

$$Y(B) = \begin{cases} -1, & \sum_{\boldsymbol{x_i} \in B} y(\boldsymbol{x_i}) = -|B| \\ +1, & otherwise \end{cases} \tag{22}$$

where B is a test bag; $Y(B)$ denotes the predicted label of B; $|B|$ is the number of instances in B. B is predicted as negative only if all instances in B are classified as negative, i.e. $\sum_{\boldsymbol{x_i} \in B} y(\boldsymbol{x_i}) = -|B|$. Otherwise, B is classified as positive.

4. Experiments

We perform experiments on real-world datasets to evaluate the effectiveness of STMIL-SVM. All experiments are run on a laptop with 2.9 GHz processor and 8 GB RAM. The objectives of our experiments are as follows:

(1) to evaluate the effectiveness of STMIL-SVM compared with state-of-the-art methods.
(2) to evaluate the sensitivity of STMIL-SVM to the labeling noise with respect to classification accuracy.
(3) to evaluate the performance variation of STMIL-SVM with different instances in instance bag, and the running time of the STMIL-SVM method.

4.1. Different methods to generate bags

In the field of machine learning, there are many clustering algorithms, such as k-means [73], EM clustering algorithm [74], DBSCAN [75]. The k-means algorithm is one of the most widely used partition-based clustering algorithms. It divides n objects into k clusters so that the clusters have higher similarity. The calculation of the similarity is performed based on the average of the objects in one cluster. It is a classic algorithm for solving clustering problems, which is simple and fast. For processing large data sets, the algorithm maintains scalability and efficiency. When the cluster is close to the Gaussian distribution, it works better. At the same time, the selection of the k value is difficult to estimate. The choice of the initial cluster center has impact on the clustering results. When the amount of data is very large, the time cost of this algorithm is very large. The EM clustering algorithm is calculated alternately in two steps. The E step is to calculate the expectation. The current estimate of the hidden variable is used to calculate its maximum likelihood estimate. The M step is the maximum likelihood value obtained on the E step. Calculate the value of the parameter. The parameters found on M step are used in the next E step calculation, which is alternated. DBSCAN is a density-based clustering algorithm. Different from the partitioning and hierarchical clustering methods, it defines the cluster as the largest set of points with density, and divides the area with sufficient high density into clusters.

There are also many image segmentation algorithms, such as GrabCut [76], MILCut [70]. GrabCut is a segmentation technique for color images and often used for human body segmentation [77]. It is an interactive iterative process that continuously performs segmentation estimation and model parameter learning. The GrabCut technique iteratively updates a three map profile that is initialized according to the results of the scan detector. MILCut is a method proposed on the basis of MIL, and is used to solve the problem of interactive image segmentation. It sets a bounding box, the target object is placed inside the bounding box, and the background is outside the bounding box. That is, an object within the bounding box is considered to be a positive bag, and an object outside the bounding box is considered to be a negative bag. Thus, the image segmentation problem is transformed into a MIL problem.

In our experiments, we will use the above three types of clustering algorithms (i.e., k-means [66], EM clustering [67] and DBSCAN [68]) to process images, and use two types of segmentation algorithms (i.e., GrabCut [69] and MILCut [70]) to cut images to get different bags, respectively.

4.2. Baselines and experimental setting

Since the proposed STMIL is MIL method, we compare its performance with state-of-the-art MIL methods as follows.

(1) The first is GMI–SVM [9], which focuses on enhancing the adaptability of traditional SVM.
(2) The second is MI–SVM [52], which trains the classifier iteratively until each positive bag has at least one instance classified as positive. It is seen that MI–SVM aims to obtain higher training accuracy of positive bags.
(3) The third is DD-SVM [78], which maps a bag of instances into a bag-level vector and uses these vectors to train a bag-level classifier, such that all points in positive bags are able to contribute to the prediction.
(4) The fourth is WellSVM [79], which is used to solve the learning problem of weakly labeled data by training the tags of incomplete examples.
(5) The fifth is PSVM-2V [80], which puts two complementary data in two views, and compensates for the gap between them.

4.3. Datasets and parameter settings

The first dataset we found is real-world *NUS-WIDE*[1] dataset [81] which is created by the Media Search Laboratory of National University of Singapore. This dataset includes 5018 unique tags from *Flickr* 269648 images and associated tags. NUS-WIDE dataset is six types of low-level features extracted from these images, including 64-D color histogram, 144-D color correlation graph, 73-D edge direction histogram, 128-D wavelet texture, 225-D block color moment and SIFT based 500-D packet and can be used to evaluate the verification of the 81 concepts.

The second dataset we found is *Flickr30k Entities*[2] dataset [82] which has become a standard benchmark for sentence-based image description. The Flickr30k dataset not only consists of 31783 photographs of everyday activities, events and scenes and 158915 captions, but also contains and extends Hodosh and others's corpus of 8092 images. Each image in the Flickr30k is described independently by five annotators, and different annotators use different levels of specificity (as shown in Fig. 2) [83].

We put the images into the A-view to create the data of A-view, and put the text descriptions into the B-view to create the data of B-view. Our target is to train two classifiers in A-view and B-view, respectively. In this paper, we recognize an image as a "bag". We regard all comments below an image as a bag, and then we utilize word segmentation method to deal with all the comments and obtain tags.

For the parameter setting of the baselines, we set the parameters similar with their work. GMI–SVM [9], MI–SVM [84], DD-SVM [85] and WellSVM [86] all use the Gaussian kernel. DD-SVM sets the regularization parameter $C = 1$. For WellSVM, C_2 is randomly selected from $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1\}$, and $C_1 = 1$. GMI–SVM sets the regularization parameter $C = 1$, and $k = \lfloor (T/15) \rfloor$ in the k-means clustering, where T is the total number of relevant images. Similar to them, the proposed STMIL-SVM uses the Gaussian kernel, set $C = 1$. In the experiments, D_A and D_B are selected from $\{0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1\}$, we uses k-means, EM clustering, DBSCAN, GrabCut and MILCut to generate bags.

4.4. Performance comparison

In this experiment, we use accuracy to measure the performance of experimental results. We use NUS-WIDE and Flickr30k Entities datasets to train classifiers in different ways and compare their performance.

Firstly, we use the k-means clustering algorithm for all bags and instances. Euclidean distance can be used for distance calculation problems in any space. Data points can exist in any space, so Euclidean distance is a more viable option. In this experiments, we use k-means clustering algorithm based on Euclidean distance method [73,87–89]. According to [9], we can give the following definition:

$$d(\boldsymbol{x_i}, \boldsymbol{x_j}) = \sqrt{\lambda^2 \|v_i - v_j\|^2 + \|t_i - t_j\|^2} \tag{23}$$

where v_i, t_i and v_j, t_j are the visual and textual features of the ith and jth image, respectively. And $d(\boldsymbol{x_i}, \boldsymbol{x_j})$ is the distance between the ith and jth image. According to the experience of the predecessors, we set $k = \lfloor (T/15) \rfloor$ in this method, and where T

[1] http://lms.comp.nus.edu.sg/research/NUS-WIDE.htm
[2] http://shannon.cs.illinois.edu/DenotationGraph/

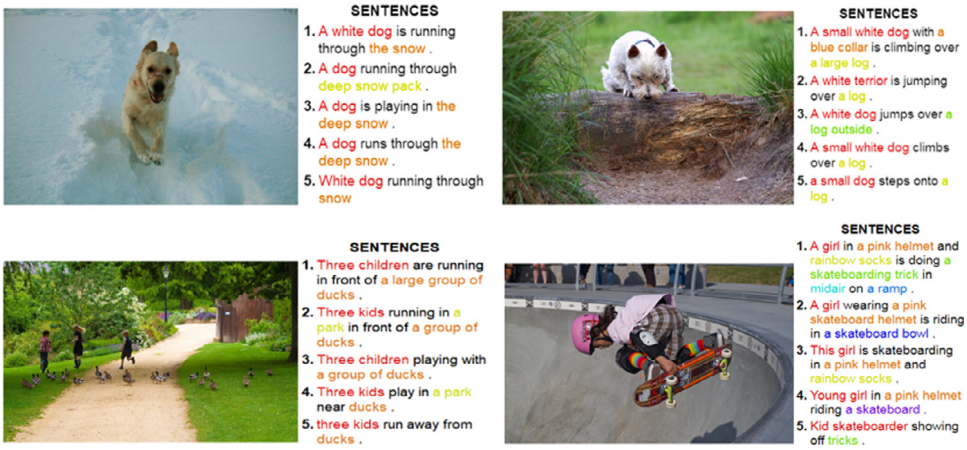

Fig. 2. Flickr30k dataset.

is the total number of images or tags. For the GMI–SVM, MI–SVM and DD-SVM baselines, since they were proposed for one-view data, we combine the extracted A-view and B-view data from the image and text to combine one vector, and conduct the baselines on the data. We compare the performance of our method with other classifiers.

Table 1 shows the results of the k-means algorithm used on the NUS-WIDE dataset and the Flickr30k Entities dataset by different methods. The k-means used here is based on Euclidean distance, and k is set 4 in the experiments. From the results of different methods, we find that the performance of the STMIL-SVM is higher than other methods on the NUS-WIDE and Flickr30k Entities dataset respectively.

Secondly, we use the EM clustering algorithm for all bags and instances for image data, respectively. For a sample set $S = \{\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_n}\}$, which consists of n independent samples. The category y_i corresponding to each sample $\boldsymbol{x_i}$ is unknown. That is, y_i is an implicit variable. Initialize the distribution parameter θ. The E step and M step are defined as follows:

$$Q_i(y_i) = P(y_i|\boldsymbol{x_i}; \theta) \tag{24}$$

$$\theta = \arg\max_{\theta} \sum_i \sum_{y_i} Q_i(y_i) \log \frac{P(\boldsymbol{x_i}, y_i; \theta)}{Q_i(y_i)} \tag{25}$$

where (24) is the E step and (25) is the M step. $Q_i(y_i)$ is the probability density function of the random variable y_i. $P(\cdot)$ is the probability density. The target is to find the value of θ. After initializing the distribution parameter θ, the loop performs the E and M steps until it converges. Its convergence conditions are as follows:

$$\begin{aligned} L(\theta^{t+1}) &\geq \sum_i \sum_{y_i} Q_i^t(y_i) \log \frac{P(\boldsymbol{x_i}, y_i; \theta^{t+1})}{Q_i^t(y_i)} \\ &\geq \sum_i \sum_{y_i} Q_i^t(y_i) \log \frac{P(\boldsymbol{x_i}, y_i; \theta^{t})}{Q_i^t(y_i)} \\ &= L(\theta^t) \end{aligned} \tag{26}$$

As shown in Table 2, the classification results of different methods are presented, the performance of the STMIL-SVM is higher than other methods. We discover that performance of different methods on the data generated by EM algorithm is higher than that by the k-means algorithm.

Thirdly, the DBSCAN is a density-based clustering algorithm, which can be applied to both convex sample sets and non-convex sample sets. DBSCAN defines the cluster as the largest set of points connected by density, divides the area with sufficient high density into clusters, and finds clusters of arbitrary shapes in the spatial database of noise. In this experiment, given a data set of images $D = \{\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_n}\}$, and $\boldsymbol{x_i}$ in D denotes an image. The radius e is initialized. An unprocessed image in the image set is selected. If the selected image is a core point, then find all objects that are reachable from the image density to form a cluster. If the selected image is an edge point (non-core object), the image is not processed. Its convergence condition is that all images are traversed.

As shown in Table 3, the classification results of different methods are listed, we find that the performance of the STMIL-SVM is higher than other methods. Furthermore, the performance of different methods on the data generated by DBSCAN algorithm is higher than that by the k-means and EM clustering algorithm respectively.

Fourthly, the GrabCut algorithm uses texture (color) information and boundary (contrast) information in the image to obtain better segmentation results with a small number of user interactions. The energy of image segmentation includes two aspects, which respectively reflects the region attribute (regional energy) and boundary property (boundary energy) of the image. In the experiment, we use the method in [76]. The set of pixels $\boldsymbol{z} = \{z_1, z_2, \ldots, z_n\}$ in the RGB color space is used to represent gray values of an image. A set $\tilde{\boldsymbol{\alpha}} = \{\tilde{\alpha_1}, \tilde{\alpha_2}, \ldots, \tilde{\alpha_n}\}, (0 \leq \tilde{\alpha_i} \leq 1)$ denotes opacity values of pixel. We can get:

$$\tilde{\alpha_i} = \begin{cases} 0, & background \\ 1, & foreground \end{cases} \tag{27}$$

where parameters $\tilde{\theta}$ represents foreground and background gray-level distributions, and consists of histograms of gray values: $\tilde{\theta} = \{h(\boldsymbol{z}; \tilde{\boldsymbol{\alpha}}), \tilde{\boldsymbol{\alpha}} \in (0, 1)\}$. Each image can be represented by $\boldsymbol{z}$, $\tilde{\boldsymbol{\alpha}}$ and $\tilde{\theta}$ together. We introduce Gaussian Mixture Model (GMM) [90] in place of histograms. Then, we introduce an additional vector $\tilde{\boldsymbol{k}} = \{\tilde{k_1}, \tilde{k_2}, \ldots, \tilde{k_n}\}$. According to the steps of the GrabCut algorithm, the foreground set T_F, background set T_B, unknown field set T_U, $\tilde{\alpha}$ are initialized. Then, we assign GMM components to pixels in T_U:

$$\tilde{k_n} = \arg\min_{\tilde{k_n}} D_n(\tilde{\alpha_n}, \tilde{k_n}, \tilde{\theta}, z_n) \tag{28}$$

where $D_n(\tilde{\alpha_n}, \tilde{k_n}, \tilde{\theta}, z_n) = -\log p(z_n \mid \tilde{\alpha_n}, \tilde{k_n}, \tilde{\theta}) - \log \pi(\tilde{\alpha_n}, \tilde{k_n})$, and $p(\cdot)$ is a Gaussian probability distribution. π is a mixture weighting coefficient. Next, learn parameters $\tilde{\theta}$ for data $\boldsymbol{z}$:

$$\tilde{\theta} = \arg\min_{\tilde{\theta}} U(\tilde{\alpha}, \tilde{k}, \tilde{\theta}, \boldsymbol{z}) \tag{29}$$

Table 1
Results of the k-means used on the different datasets by different methods.

	STMIL-SVM	GMI–SVM	MI–SVM	DD-SVM	WellSVM	PSVM-2V
NUS-WIDE	**66.2%**	62.6%	61.6%	59.4%	52.1%	64.9%
Flickr30k Entities	**65.9%**	61.7%	60.8%	59.2%	51.6%	62.3%

Table 2
Results of the EM clustering used on the different datasets by different methods.

	STMIL-SVM	GMI–SVM	MI–SVM	DD-SVM	WellSVM	PSVM-2V
NUS-WIDE	**68.1%**	63.9%	62.8%	62.3%	54.6%	67.4%
Flickr30k Entities	**67.8%**	63.5%	62.5%	61.4%	53.8%	65.1%

Table 3
Results of the DBSCAN used on the different datasets by different methods.

	STMIL-SVM	GMI–SVM	MI–SVM	DD-SVM	WellSVM	PSVM-2V
NUS-WIDE	**68.9%**	64.7%	63.6%	64.3%	55.9%	66.8%
Flickr30k Entities	**67.6%**	63.7%	63.4%	63.9%	54.4%	63.9%

where the data item U is used to evaluate the fit of the opacity distribution $\tilde{\alpha}$ to the data $\boldsymbol{z}$. Finally, minimum cutting method to estimate segmentation:

$$\min_{\tilde{\alpha}} \min E(\tilde{\alpha}, \tilde{k}, \tilde{\theta}, \boldsymbol{z}) \tag{30}$$

where E is an energy function of image segmentation. Repeat from (28), until convergence.

As shown in Table 4, the classification results of different methods are listed, the performance of the STMIL-SVM is still higher than other methods. The performance of different methods on the data generated by GrabCut algorithm is lower than that by the DBSCAN algorithm, and between the k-means and EM clustering algorithm.

Fifthly, MILCut uses pixels on the sweeping lines and connects the lines into a bounding box. Similar to the method in [70], we also use the superpixels method [91] to process the original image and convert it into a MIL problem. In the experiments, we introduce simple linear iterative clustering (SLIC) [92], which is an adaptation of k-means for superpixel generation. To produce roughly equally sized superpixels, thus the grid interval is $G_n = \sqrt{N/K}$, where N is the number of pixels and K is the number of superpixels. We set $N = 2400$, this means that each image produces 2400 pixels. We divide the image into slices according to the superpixel. We use a bounding box to divide the image into two regions. Objects in the bounding box are treated as a positive bags, and objects outside the bounding box are considered a negative bags. This translates into a typical MIL problem.

As shown in Table 5, the classification results of different methods are presented, we find that the proposed STMIL-SVM outperforms other methods. The performance of different methods on the data generated by GrabCut algorithm method is better than that of the GrabCut algorithm, and between the k-means and EM clustering algorithm. In addition, the performance of MILCut is higher than that by the above three clusters algorithm (i.e., k-means, EM clustering and DBSCAN).

In all, we also utilize F1-measure to compare the algorithms, and the results under different feature extraction methods for multiple instance learning are listed in Tables 6 and 7, we can observe that, the performance of the proposed STMIL-SVM method always performs better than other methods. Above, we have compared the proposed STMIL-SVM method with GMI–SVM, MI–SVM, DD-SVM, WellSVM and PSVM-2V methods based on five image features generation methods, the results show that our method can always yield a higher performance compared with other methods. In addition, the bags generated by DBSCAN method can deliver a higher performance compared with other bags generation methods.

The clustering methods (k-means, EM clustering and DBSCAN) are used to transform the image into the multiple instance bag, this is kind of data processing. In the proposed STMIL method, we do not utilize the clustering method. In addition, for the feature extraction for bag generation, we can also use the two GrabCut, MILCut, which are not clustering methods. Even for the clustering method k-means, EM clustering and DBSCAN, if the clustering result is not appropriate, the multiple instance bag method can reduce the effect of them on the subsequent learning. From Tables 1–3, we can see that the results using bag generation with k-means, EM clustering, and DBSCAN are comparable stable, similar as the results with the bag generation GrabCut, MILCut.

4.5. Variation to bag sample number

We compare the proposed STMIL-SVM with GMI–SVM, MI–SVM, WellSVM and PSVM-2V using different numbers of training bags. We set $n_B = 2, 4, 6, 8$ and 10. The results of different methods on the NUS-WIDE and Flickr30k Entities dataset are shown in Tables 8 and 9.

From the Table, we can find that as the number of bags increases, the performance of different methods also increases. In addition, the results show that our method can always yield a higher performance compared with other methods. Furthermore, the performance of our method and GMI–SVM reaches a peak when number of training bags $n_B = 8$.

4.6. Sensitivity to input data noise

Our experiments also test the sensitivity of algorithm performance to input data noise. For each dataset, we first calculate the standard deviation σ_i^0 of the entire data along the ith dimension, and then obtain the standard deviation of the Gaussian noise σ_i randomly from the range $[0, 2 \cdot \sigma_i^0]$. In this way, noise can be added to the positive class as a vector having the same dimension as the original dataset. Fig. 3 illustrates the basic idea of the method used to add the noise to data examples. Specifically, the standard deviation σ_i^0 of the entire data along the ith dimension is first obtained. In order to model the difference in noise on different dimensions, we define the standard deviation σ_i along the ith dimension, whose value is randomly drawn from the range $[0, 2 \cdot \sigma_i^0]$. Then, for the ith dimension, we add noise from a random distribution with standard deviation σ_i. In this way, a data example $\boldsymbol{x_j}$ is added with the noise, which can be presented as a vector [93].

$$\sigma^{\boldsymbol{x_j}} = [\sigma_1^{\boldsymbol{x_j}}, \sigma_2^{\boldsymbol{x_j}}, \ldots, \sigma_{n-1}^{\boldsymbol{x_j}}, \sigma_n^{\boldsymbol{x_j}}] \tag{31}$$

Table 4
Results of the GrabCut used on the different datasets by different methods.

	STMIL-SVM	GMI–SVM	MI–SVM	DD-SVM	WellSVM	PSVM-2V
NUS-WIDE	**64.7%**	60.9%	58.3%	58.2%	48.7%	61.3%
Flickr30k Entities	**63.1%**	59.8%	56.4%	57.6%	48.3%	61.8%

Table 5
Results of the MILCut used on the different datasets by different methods.

	STMIL-SVM	GMI–SVM	MI–SVM	DD-SVM	WellSVM	PSVM-2V
NUS-WIDE	**65.3%**	62.1%	60.2%	58.5%	50.2%	64.5%
Flickr30k Entities	**65.1%**	61.6%	58.7%	56.6%	49.1%	63.8%

Table 6
F1-measure values of different methods on the NUS-WIDE dataset.

	STMIL-SVM	GMI–SVM	MI–SVM	DD-SVM	WellSVM	PSVM-2V
k-means	**67.6%**	61.9%	60.5%	62.5%	51.4%	63.7%
EM Clustering	**66.8%**	64.1%	63.2%	60.0%	53.3%	65.6%
DBSCAN	**64.7%**	64.5%	62.0%	60.4%	58.4%	63.1%
GrabCut	**62.5%**	61.9%	58.8%	57.4%	52.5%	60.3%
MILCut	**65.6%**	64.0%	59.8%	58.3%	49.9%	62.4%

Table 7
F1-measure values of different methods on the Flickr30k entities dataset.

	STMIL-SVM	GMI–SVM	MI–SVM	DD-SVM	WellSVM	PSVM-2V
k-means	**63.7%**	63.2%	58.4%	60.5%	50.7%	59.7%
EM Clustering	**68.7%**	65.9%	61.5%	59.1%	55.2%	67.1%
DBSCAN	**67.8%**	65.4%	61.3%	60.7%	52.3%	66.5%
GrabCut	**65.7%**	63.6%	62.7%	58.6%	52.9%	65.3%
MILCut	**64.6%**	60.1%	57.7%	55.8%	53.8%	63.9%

Table 8
Performance varying for different number of instances in each bag on the NUS-WIDE dataset.

		STMIL-SVM	GMI–SVM	MI–SVM	WellSVM	PSVM-2V
k-means	$n_B = 2$	**66.3%**	62.8%	61.7%	52.1%	62.2%
	$n_B = 4$	**67.1%**	64.5%	63.3%	52.5%	63.8%
	$n_B = 6$	**68.3%**	66.4%	64.7%	52.9%	66.7%
	$n_B = 8$	**69.8%**	66.8%	64.7%	53.4%	67.1%
	$n_B = 10$	**69.7%**	66.6%	64.8%	53.9%	67.5%
EM Clustering	$n_B = 2$	**66.8%**	63.7%	62.1%	52.7%	62.4%
	$n_B = 4$	**67.7%**	65.5%	64.3%	53.1%	64.1%
	$n_B = 6$	**68.5%**	66.7%	65.6%	57.4%	65.8%
	$n_B = 8$	**69.3%**	66.9%	65.5%	54.6%	67.2%
	$n_B = 10$	**69.1%**	66.8%	65.3%	54.8%	68.8%
DBSCAN	$n_B = 2$	**67.3%**	64.5%	64.2%	53.8%	63.0%
	$n_B = 4$	**68.1%**	64.9%	64.7%	55.3%	64.7%
	$n_B = 6$	**68.9%**	66.4%	64.9%	53.8%	67.9%
	$n_B = 8$	**70.1%**	68.3%	67.3%	59.3%	68.2%
	$n_B = 10$	**69.5%**	67.1%	66.8%	61.2%	68.1%
MILCut	$n_B = 2$	**61.6%**	60.2%	54.3%	47.4%	60.7%
	$n_B = 4$	**63.4%**	62.3%	58.1%	48.2%	61.4%
	$n_B = 6$	**65.3%**	64.2%	60.3%	48.7%	64.9%
	$n_B = 8$	**67.9%**	66.1%	62.8%	49.6%	66.5%
	$n_B = 10$	**66.7%**	65.8%	63.1%	51.7%	66.8%
GrabCut	$n_B = 2$	**60.3%**	58.8%	57.2%	43.9%	60.1%
	$n_B = 4$	**63.0%**	61.7%	60.1%	45.1%	62.6%
	$n_B = 6$	**65.2%**	63.3%	62.4%	47.3%	64.4%
	$n_B = 8$	**66.6%**	66.2%	63.1%	49.6%	65.3%
	$n_B = 10$	**65.9%**	65.1%	62.3%	50.5%	65.7%

where n denotes the number of dimensions for a data example $\boldsymbol{x_j}$, and $\sigma_i^{\boldsymbol{x_j}}, i = 1, \ldots, n$ represents the noise added into the ith dimension of the data example.

In our experiments, we make the percentage of the data noise vary from 0% to 30%. Here, we utilize the data generated by vector plus offset constant method as an example, and add the noise into the instances of each bag. The Figs. 4 and 5 illustrates the effect of different proportions of noise on performance. From the two figures, we find that as the level of noise increases, the performance of all methods decreases. However, STMIL-SVM can always obtain a higher accuracy and is less sensitive to noise compared with other methods.

4.7. Parameter sensitivity analysis

In this set of experiment, we test the sensitivity of the STMIL-SVM to its parameters using the NUS-WIDE data set as example. In the proposed method, there are several parameters, we analyze the performance variation at different values of parameters. We

Table 9
Performance varying for different number of instances in each bag on the Flickr30k Entities dataset.

		STMIL-SVM	GMI–SVM	MI–SVM	WellSVM	PSVM-2V
k-means	$n_B = 2$	**65.9%**	61.7%	60.8%	51.7%	62.2%
	$n_B = 4$	**66.4%**	63.4%	62.9%	51.9%	63.7%
	$n_B = 6$	**67.6%**	65.1%	64.1%	53.6%	65.9%
	$n_B = 8$	**69.1%**	66.6%	64.2%	53.8%	67.4%
	$n_B = 10$	**68.9%**	66.3%	64.1%	54.7%	68.1%
EM Clustering	$n_B = 2$	**66.3%**	62.8%	58.1%	50.6%	62.7%
	$n_B = 4$	**66.9%**	63.3%	59.2%	51.5%	62.9%
	$n_B = 6$	**67.8%**	64.7%	62.5%	52.4%	63.3%
	$n_B = 8$	**69.4%**	65.1%	63.7%	53.1%	65.5%
	$n_B = 10$	**69.3%**	65.1%	63.5%	54.3%	66.1%
DBSCAN	$n_B = 2$	**66.9%**	63.3%	60.4%	52.2%	62.4%
	$n_B = 4$	**67.4%**	64.7%	63.6%	54.7%	63.2%
	$n_B = 6$	**67.9%**	65.8%	64.7%	55.5%	65.6%
	$n_B = 8$	**68.7%**	67.2%	66.6%	58.1%	66.5%
	$n_B = 10$	**68.6%**	67.0%	66.3%	56.8%	67.2%
MILCut	$n_B = 2$	**60.1%**	59.8%	55.6%	46.2%	58.3%
	$n_B = 4$	**62.3%**	62.0%	57.7%	47.6%	61.8%
	$n_B = 6$	**66.2%**	63.1%	58.1%	48.1%	63.4%
	$n_B = 8$	**66.5%**	65.7%	58.8%	49.9%	64.9%
	$n_B = 10$	**66.1%**	63.9%	60.1%	50.3%	65.2%
GrabCut	$n_B = 2$	**58.7%**	56.5%	56.1%	44.8%	57.3%
	$n_B = 4$	**62.2%**	60.4%	60.0%	45.3%	59.1%
	$n_B = 6$	**65.0%**	63.1%	62.7%	46.2%	62.5%
	$n_B = 8$	**66.1%**	65.3%	64.1%	48.1%	64.8%
	$n_B = 10$	**64.8%**	64.1%	61.2%	49.9%	63.2%

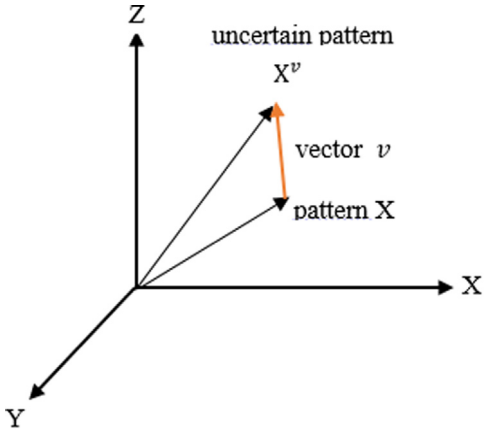

Fig. 3. Add the noise to a data example.

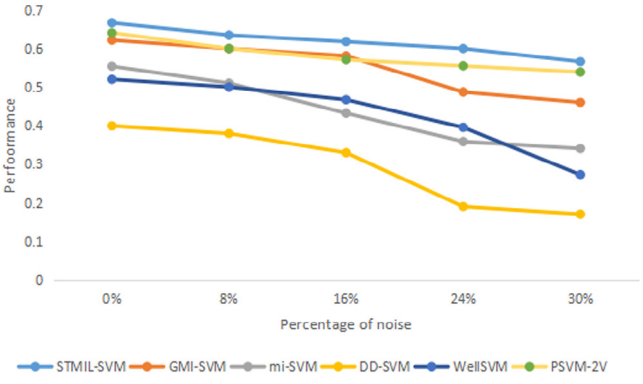

Fig. 4. NUS-WIDE dataset.

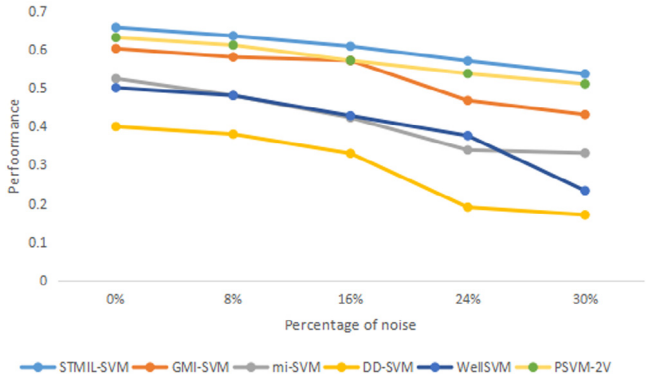

Fig. 5. Flickr30k Entities dataset.

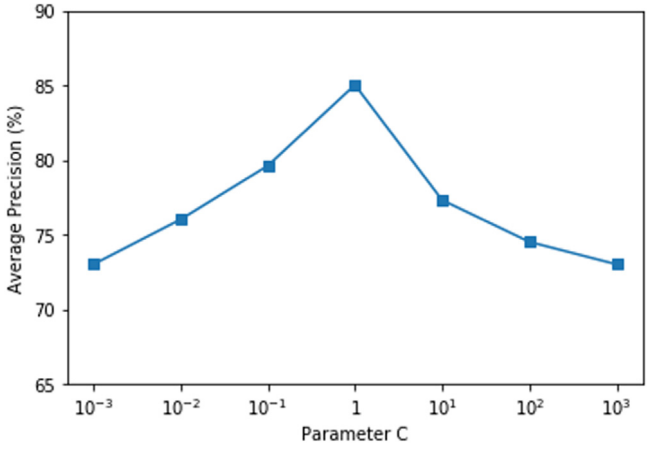

Fig. 6. The Parameter Sensitiveness of C on NUS-WIDE.

first concern the four parameters C, D_A, D_B for parameter sensitivity analysis in STMIL-SVM. We first draw the figure of parameter C influence when D_A and D_B are set as a fixed value, in which C is changed from 10^{-3} to 10^3. The result is shown in Fig. 6, we discover that the performance of STMIL-SVM increases as the C value increases from 10^{-3}, and the performance reaches its

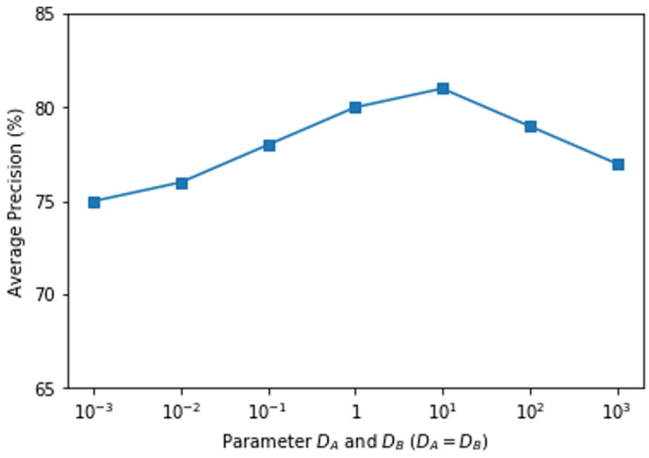

Fig. 7. The Parameter Sensitiveness of D_A and D_B on NUS-WIDE.

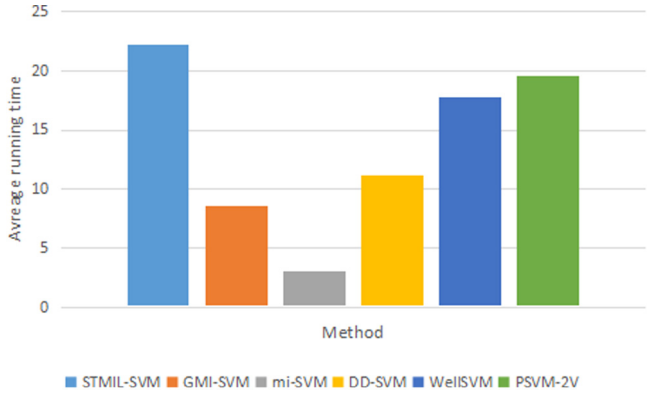

Fig. 8. NUS-WIDE dataset.

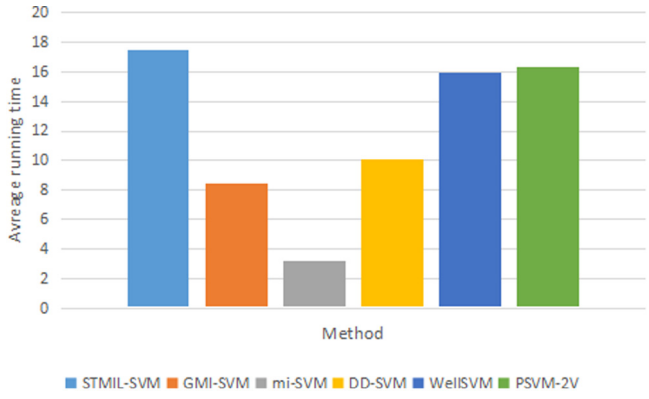

Fig. 9. Flickr30k Entities dataset.

peak value at $C = 1$, and then the performance decreases as the C value continues to increase. In addition, D_A and D_B are the parameters to balance the importance of the samples. In order to study the influence of the values of D_A and D_B on the performance of the STMIL-SVM, we let $D_A = D_B$ and illustrate the performance variation as the D_A and D_B change. The result is illustrated in Fig. 7, in which D_A and D_B increase from 10^{-3} to 10^{3}. From the figure, we find that the performance of the proposed method increases to its peak and then decreases as D_A and D_B increase.

4.8. Average running time comparison

We have compared the performance of our method and other methods, and it is still interesting to know the average running time of each SVM algorithm. All the experiments are implemented with MATLAB codes. Figs. 8 and 9 illustrates the average running time of different methods executing the NUS-WIDE and Flickr30k Entities dataset, respectively. The average runtime of STMIL-SVM is more than that of other methods, because our method needs to train two classifiers simultaneously. In addition, the average time consumed by the STMIL-SVM and WellSVM methods is roughly the same because both methods are based on two-view learning.

5. Conclusion and future work

In this paper, we propose a new similarity-based two-view MIL (STMIL). We first utilize the two-view learning method to place images and text information in two views, and convert the image classification problem into a MIL problem. We then propose the two-view MIL for image and textual classification. We present the original STMIL-SVM problem and solve it using the Lagrangian method, and then present the optimization framework of STMIL-SVM method. In the experimental part, we use three clustering algorithms (k-means, EM clustering and DBSCAN) and two image segmentation algorithms (GrabCut and MILCut) to process the images and compare their performance. Experiments have shown that our method has higher performance compared with other MIL methods.

In the future, we will expand our approach and framework to the field of video processing and online data.

CRediT authorship contribution statement

Yanshan Xiao: Conceptualization, Methodology, Investigation. **Zijian Yin:** Methodology, Validation, Visualization.

Acknowledgments

This work was supported in part by the Natural Science Foundation of China under Grant 61876044 and Grant 61672169, in part by the NSFC-Guangdong Joint Found U1501254, in part by Guangdong Natural Science Foundation under Grant 2020A1515010670 and 2020A1515011501.

References

[1] Thomas G. Dietterich, Richard H. Lathrop, Tomás Lozano-Pérez, Solving the multiple instance problem with axis-parallel rectangles, Artificial Intelligence 89 (1) (1997) 31–71, http://dx.doi.org/10.1016/S0004-3702(96)00034-3.

[2] Qingping Tao, Stephen Scott, N.V. Vinodchandran, Thomas Takeo Osugi, Brandon Mueller, An extended kernel for generalized multiple-instance learning, in: IEEE International Conference on TOOLS with Artificial Intelligence, 2004, pp. 272–277.

[3] Boris Babenko, Ming Hsuan Yang, Serge Belongie, Robust Object Tracking with Online Multiple Instance Learning, IEEE Computer Society, 2011, pp. 1619–1632.

[4] Qi Zhang, Sally A. Goldman, Wei Yu, Jason Fritts, Content-based image retrieval using multiple-instance learning, in: Nineteenth International Conference on Machine Learning, 2002, pp. 682–689.

[5] Dan Zh, Fei W, Zh S, Changshui Zh, Interactive localized content based image retrieval with multiple-instance active learning, Pattern Recognit. 43 (2) (2010) 478–484.

[6] Daxiang Li, Xiaoqiang Zhao, Weihua Liu, Na Li, A novel MIL algorithm for image classification, in: International Conference on Electronics, Communications and Control, 2012, pp. 3221–3225.

[7] Zhi Hua Zhou, Multi-instance Learning: A Survey, Department of Computer Science Technology, 2004.

[8] Peter A. Flach, Adam Kowalczyk, Alex J. Smola, Multi-instance kernels, in: Nineteenth International Conference on Machine Learning, 2002, pp. 179–186.

[9] L. Duan, W. Li, I.W. Tsang, D. Xu, Improving web image search by bag-based reranking, IEEE Trans. Image Process. 20 (11) (2011) 3280–3290.

[10] Yingying Wang, Chun Zhang, Zhihua Wang, Rate distortion multiple instance learning for image classification, in: IEEE International Conference on Image Processing, 2013, pp. 3235–3238.

[11] Yingjie Tian, X.U. Dongkuan, Chunhua Zhang, A review of multi-instance learning research, Oper. Res. Trans. (2018).

[12] Kajsa M, Jon Yngve H, Fred G, A bag-to-class divergence approach to multiple-instance learning, 2018.

[13] C. Zhang, J.C. Platt, P.A. Viola, Multiple instance boosting for object detection, in: International Conference on Neural Information Processing Systems, 2005.

[14] Fu Zhouyu, Robles Kelly Antonio, Zhou Jun, MILIS: multiple instance learning with instance selection, IEEE Trans. Pattern Anal. Mach. Intell. 33 (5) (2011) 958–977.

[15] Jia Wu, Shirui Pan, Xingquan Zhu, Chengqi Zhang, Xindong Wu, Multi-instance learning with discriminative bag mapping, IEEE Trans. Knowl. Data Eng. PP (99) (2018) 1.

[16] Shafin Rahman, Salman Khan, Deep multiple instance learning for zero-shot image tagging, 2018.

[17] L. Yuan, X. Wen, L. Zhao, H. Xu, An iterative instance selection based framework for multiple-instance learning, in: 2018 IEEE 30th International Conference on Tools with Artificial Intelligence, ICTAI, 2018, pp. 772–779, http://dx.doi.org/10.1109/ICTAI.2018.00121.

[18] Anit V. Manjaly, B. Shanmuga Priya, Malayalam text and non-text classification of natural scene images based on multiple instance learning, in: IEEE International Conference on Advances in Computer Applications, 2017, pp. 190–196.

[19] Bo Liu, Yanshan Xiao, Zhifeng Hao, A selective multiple instance transfer learning method for text categorization problems, Knowl.-Based Syst. 141 (2018).

[20] Wei He, Yu Wang, Text Representation and Classification Based on Multi-Instance Learning, 2009, pp. 34–39.

[21] J. Wu, Yinan Yu, Chang Huang, Kai Yu, Deep multiple instance learning for image classification and auto-annotation, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2015, pp. 3460–3469, http://dx.doi.org/10.1109/CVPR.2015.7298968.

[22] X. Liu, L. Jiao, J. Zhao, J. Zhao, D. Zhang, F. Liu, S. Yang, X. Tang, Deep multiple instance learning-based spatial-spectral classification for PAN and MS imagery, IEEE Trans. Geosci. Remote Sens. 56 (1) (2018) 461–473, http://dx.doi.org/10.1109/TGRS.2017.2750220.

[23] M. Liu, J. Zhang, E. Adeli, D. Shen, Landmark-based deep multi-instance learning for brain disease diagnosis, Med. Image Anal. 43 (2017) 157.

[24] D. Varga, T. Szórényi, Person re-identification based on deep multi-instance learning, in: 2017 25th European Signal Processing Conference, EUSIPCO, 2017, pp. 1559–1563, http://dx.doi.org/10.23919/EUSIPCO.2017.8081471.

[25] Yanshan Xiao, Bo Liu, Zhifeng Hao, Longbing Cao, A similarity-based classification framework for multiple-instance learning, IEEE Trans. Cybern. 44 (4) (2014) 500.

[26] In Cowan, G. Tesauro, Virginia R. De Sa, Learning classification with unlabeled data, in: Advances in Neural Information Processing Systems, 1993, pp. 112–119.

[27] Zhe Wang, Yiwen Zhu, Zhaozhi Chen, Jing Zhang, Wenli Du, Multi-view learning with fisher kernel and bi-bagging for imbalanced problem, Appl. Intell. 49 (2019) 3109–3122.

[28] Xie, Xijiong, Regularized multi-view least squares twin support vector machines, Appl. Intell. 48 (2018) 3108–3115.

[29] Avrim Blum, Combining labeled and unlabeled data with co-training, in: Conference on Computational Learning Theory, 1998, pp. 92–100.

[30] Alain Rakotomamonjy, Francis R. Bach, Stéphane Canu, Yves Grandvalet, Simplemkl, J. Mach. Learn. Res. 9 (3) (2008) 2491–2521.

[31] Francis R. Bach, Consistency of the group lasso and multiple kernel learning, J. Mach. Learn. Res. 9 (2) (2007) 1179–1225.

[32] Francis R. Bach, Gert R.G. Lanckriet, Michael I. Jordan, Multiple kernel learning, conic duality, and the SMO algorithm, in: International Conference, 2004, pp. 6.

[33] S. Molina-Giraldo, A.M. Alvarez-Meza, D.H. Peluffo-Ordo?ez, G. Castellanos-Dominguez, Image Segmentation Based on Multi-Kernel Learning and Feature Relevance Analysis, 2012, pp. 501–510.

[34] Lining Zhang, Lipo Wang, Weisi Lin, Biased subspace learning for SVM relevance feedback in content-based image retrieval, Commun. Signal Process. (2011) 1–5.

[35] Duc Son Pham, Svetha Venkatesh, Supervised subspace learning with multi-class Lagrangian SVM on the Grassmann manifold, 7106 (2011) 241–250.

[36] Meng Chen, Lin Lin Zhang, Xiaohui Yu, Yang Liu, Weighted co-training for cross-domain image sentiment classification, 32 (4) (2017) 714–725.

[37] Xiangrong Zhang, Qiang Song, Ruochen Liu, Wenna Wang, Licheng Jiao, Modified co-training with spectral and spatial views for semisupervised hyperspectral image classification, IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens. 7 (6) (2014) 2044–2055.

[38] Yingchao Xiao, Huangang Wang, Lin Zhang, Wenli Xu, Two methods of selecting Gaussian kernel parameters for one-class SVM and their application to fault detection, Knowl.-Based Syst. 59 (2) (2014) 75–84.

[39] S. Chanda, S. Pal, U. Pal, Word-wise Sinhala Tamil and English script identification using Gaussian kernel SVM, in: International Conference on Pattern Recognition, 2008, pp. 1–4.

[40] H.U. Xiangping, Multiple feature fusion via multiple kernel learning for image classification, Comput. Eng. Appl. (2016).

[41] X.Z. Qi, Q. Wang, An image classification approach based on sparse coding and multiple kernel learning, Acta Electron. Sin. 40 (4) (2012) 773–779.

[42] Jun Yu, Feng Lin, Hock Soon Seah, Cuihua Li, Ziyu Lin, Image classification by multimodal subspace learning, Pattern Recognit. Lett. 33 (9) (2012) 1196–1204.

[43] Xiaozhao Fang, Shaohua Teng, Zhihui Lai, Zhaoshui He, Shengli Xie, Wai Keung Wong, Robust latent subspace learning for image classification, IEEE Trans. Neural Netw. Learn. Syst. PP (99) (2017) 1–14.

[44] Qiu Xiao, Jianhua Dai, Jiawei Luo, Hamido Fujita, Multi-view manifold regularized learning-based method for prioritizing candidate disease miRNAs, Knowl.-Based Syst. 175 (2019) http://dx.doi.org/10.1016/j.knosys.2019.03.023.

[45] Xin Shu, Peisen Yuan, Haiyan Jiang, Darong Lai, Multi-view uncorrelated discriminant analysis via dependence maximization, Appl. Intell. 49 (2) (2019) 650–660.

[46] Yiling Zhang, Yan Yang, Tianrui Li, Hamido Fujita, A multitask multiview clustering algorithm in heterogeneous situations based on LLE and LE, Knowl.-Based Syst. 163 (2018) http://dx.doi.org/10.1016/j.knosys.2018.10.001.

[47] Hao Wang, Yan Yang, Bing Liu, Hamido Fujita, A study of graph-based system for multi-view clustering, Knowl.-Based Syst. 163 (2018) http://dx.doi.org/10.1016/j.knosys.2018.10.022.

[48] Jason D.R. Farquhar, David R. Hardoon, Hongying Meng, John Shawe-Taylor, Sándor Szedmák, Two view learning: SVM-2K, theory and practice, Adv. Neural Inf. Process. Syst. (2005) 355–362.

[49] Guangxia Li, Steven C.H. Hoi, Kuiyu Chang, Two-view transductive support vector machines, in: Siam International Conference on Data Mining, SDM 2010, April 29 - May 1, 2010, Columbus, Ohio, Usa, 2010, pp. 235–244.

[50] Sheng Wang, Jianfeng Lu, Xingjian Gu, Chunhua Shen, Rui Xia, Jingyu Yang, Canonical principal angles correlation analysis for two-view data, J. Vis. Commun. Image Represent. 35 (C) (2016) 209–219.

[51] Hongying Meng, B. Romera-Paredes, N. Bianchi-Berthouze, Emotion recognition by two view SVM-2K classifier on dynamic facial expression features, in: IEEE International Conference on Automatic Face Gesture Recognition and Workshops, 2011, pp. 854–859.

[52] Stuart Andrews, Ioannis Tsochantaridis, Thomas Hofmann, Support vector machines for multiple-instance learning, Adv. Neural Inf. Process. Syst. 15 (2) (2003) 561–568.

[53] Oded Maron, Tomás Lozano-Pérez, A framework for multiple-instance learning, Adv. Neural Inf. Process. Syst. 200 (2) (1998) 570–576.

[54] Jun Wang, Jean Daniel Zucker, Solving the multiple-instance problem: A lazy learning approach, in: Seventeenth International Conference on Machine Learning, 2000, pp. 1119–1126.

[55] Qi Zhang, Sally A. Goldman, EM-DD: an improved multiple-instance learning technique, in: International Conference on Neural Information Processing Systems: Natural and Synthetic, 2001, pp. 1073–1080.

[56] Stephen Scott, Jun Zhang, Joshua Brown, On generalized multiple-instance learning, IEEE Trans. Pattern Anal. Mach. Intell. 30 (12) (2008) 2084–2098.

[57] Annabella Astorino, Antonio Fuduli, Manlio Gaudioso, Eugenio Vocaturo, A multiple instance learning algorithm for color images classification, in: Proceedings of the 22nd International Database Engineering & Applications Symposium, IDEAS 2018, Villa San Giovanni, Italy, June 18-20, 2018, 2018, pp. 262–266, http://dx.doi.org/10.1145/3216122.3216144, url=http://doi.acm.org/10.1145/3216122.3216144.

[58] Xizhan Gao, Quansen Sun, Haitao Xu, Multiple instance learning via semi-supervised Laplacian TSVM, Neural Process. Lett. 46 (1) (2017) 1–14.

[59] Joana Correia, Isabel Trancoso, Bhiksha Raj, Adaptation of SVM for MIL for inferring the polarity of movies and movie reviews, in: Spoken Language Technology Workshop, 2017, pp. 258–264.

[60] T. Lai, H. Fujita, C. Yang, Q. Li, R. Chen, Robust model fitting based on greedy search and specified inlier threshold, IEEE Trans. Ind. Electron. 66 (10) (2019) 7956–7966, http://dx.doi.org/10.1109/TIE.2018.2881950.

[61] T. Lai, R. Chen, C. Yang, Q. Li, H. Fujita, A. Sadri, H. Wang, Efficient robust model fitting for multistructure data using global greedy search, IEEE Trans. Cybern. (2019) 1–13, http://dx.doi.org/10.1109/TCYB.2019.2900096.

[62] Maximilian Ilse, Jakub M. Tomczak, Max Welling, Attention-based Deep Multiple Instance Learning, 2018.

[63] Lei Zhou, Yu Zhao, Jie Yang, Qi Yu, Xun Xu, Deep multiple instance learning for automatic detection of diabetic retinopathy in retinal images, IET Image Process. 12 (4) (2018) 563–571.
[64] T. Yu, M. Wang, Y. Lv, L. Xue, J. Liu, Interpretative topic categorization via deep multiple instance learning, in: 2018 IJCNN, 2018, pp. 1–7, http://dx.doi.org/10.1109/IJCNN.2018.8489395.
[65] Qilong Li, Xiaohong Wang, Image classification based on SIFT and SVM, in: 2018 IEEE/ACIS 17th International Conference on Computer and Information Science, ICIS, 2018.
[66] P. Jeyanthi, V. Jawahar, Senthil Kumar, Image classification by K-means clustering, in: EM and Normalized Cuts, 219, Department of EECS, 2010, pp. 91–99.
[67] P.R. Kersten, Jong Sen Lee, T.L. Ainsworth, Unsupervised classification of polarimetric synthetic aperture radar images using fuzzy clustering and EM clustering, IEEE Trans. Geosci. Remote Sens. 43 (3) (2005) 519–527.
[68] Anirudh Harisinghaney, Aman Dixit, Saurabh Gupta, Anuja Arora, Text and image based spam email classification using KNN, Naive Bayes and Reverse DBSCAN algorithm, in: International Conference on Optimization, Reliabilty, and Information Technology, 2014, pp. 153–155.
[69] Yong Zhang, Jiazheng Yuan, Hongzhe Liu, Qing Li, GrabCut image segmentation algorithm based on structure tensor, J. China Univ. Posts Telecommun. 24 (2) (2017) 38–47.
[70] Jiajun W, Yibiao Zhao, J Yan Zhu, Siwei L, Zhuowen T, MILCut: A sweeping line multiple instance learning paradigm for interactive image segmentation, in: IEEE Conference on CVPR, 2014, pp. 256–263.
[71] Guifen Zhao, Yanjun Liu, Zhang Wei, Yiou Wang, TFIDF based feature words extraction and topic modeling for short text, in: The 2018 2nd International Conference, 2018.
[72] Qin Zhang, Yingjie Tian, Dalian Liu, Nonparallel support vector machines for multiple-instance learning, Procedia Comput. Sci. 17 (2013) 1063–1072.
[73] J.A. Hartigan, A K-means clustering algorithm, Appl. Stat. 28 (1) (1979) 100–108.
[74] Hans Peter Kriegel, Alexey Pryakhin, Matthias Schubert, An EM-approach for clustering multi-instance objects, in: Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, 2006, pp. 139–148.
[75] Martin Ester, Hans Peter Kriegel, Xiaowei Xu, A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise, in: International Conference on Knowledge Discovery and Data Mining, 1996, pp. 226–231.
[76] Carsten Rother, Vladimir Kolmogorov, Andrew Blake, "GrabCut": interactive foreground extraction using iterated graph cuts, in: ACM SIGGRAPH, 2004, pp. 309–314.
[77] Varun Gulshan, Victor Lempitsky, Andrew Zisserman, Humanising GrabCut: Learning to segment humans using the Kinect, in: IEEE International Conference on Computer Vision Workshops, 2011, pp. 1127–1133.
[78] Y. Chen, Image categorization by learning and reasoning with regions, J. Mach. Learn. Res. 5 (4) (2004) 913–939.
[79] Yu Feng Li, Ivor W. Tsang, James T. Kwok, Zhi Hua Zhou, Convex and scalable weakly labeled SVMs, J. Mach. Learn. Res. 14 (1) (2013) 2151–2188.
[80] J. Tang, Y. Tian, P. Zhang, X. Liu, Multiview privileged support vector machines, IEEE Trans. Neural Netw. Learn. Syst. 29 (8) (2018) 3463–3477, http://dx.doi.org/10.1109/TNNLS.2017.2728139.
[81] Tat Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, Yantao Zheng, NUS-WIDE:a real-world web image database from National University of Singapore, in: ACM International Conference on Image and Video Retrieval, 2009, pp. 1–9.
[82] Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models, in: IEEE International Conference on Computer Vision, 2015, pp. 2641–2649.
[83] P. Y, A. L, M. H, J. H, From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions, Nlp.cs.illinois.edu, 2014.
[84] Zhiyang Chen, Liya Huang, Yangyang Shen, Jun Wang, Ruijie Zhao, Jiafei Dai, A new algorithm for classification of ictal and pre-ictal epilepsy ECoG using MI and SVM, in: 2017 International Conference on Signals and Systems, ICSigSys, 2017, pp. 212–216, http://dx.doi.org/10.1109/ICSIGSYS.2017.7967043.
[85] Fanyong Cheng, Zhang Jing, Zuoyong Li, Mingzhu Tang, Double distribution support vector machine, Pattern Recognit. Lett. 88 (C) (2017) 20–25.
[86] Xinxing Xu, Wen Li, Dong Xu, Ivor W. Tsang, Co-labeling for multi-view weakly labeled learning, IEEE Trans. Pattern Anal. Mach. Intell. 38 (6) (2016) 1113–1125.
[87] T. Kanungo, D.M. Mount, N.S. Netanyahu, C.D. Piatko, R. Silverman, A.Y. Wu, An efficient k-means clustering algorithm: analysis and implementation, IEEE Trans. Pattern Anal. Mach. Intell. 24 (7) (2002) 881–892.
[88] Anil K. Jain, Data Clustering: 50 Years beyond K-Means, Springer Berlin Heidelberg, 2008, pp. 651–666.
[89] Georg Peters, Some refinements of rough K-means, Pattern Recognit. 39 (8) (2006) 1481–1491.
[90] Vladimir Kolmogorov, Ramin Zabih, What energy functions can be minimized via graph cuts, in: ECCV, 2002, pp. 65–81.
[91] Xiaofeng Ren, Jitendra Malik, Learning a classification model for segmentation, in: Iccv, vol. 1, 2003, pp. 10–17.
[92] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, S. Ssstrunk, SLIC superpixels compared to state-of-the-art superpixel methods, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2274–2282.
[93] Bo Liu, Yanshan Xiao, Philip S. Yu, Zhifeng Hao, Longbing Cao, An efficient approach for outlier detection with imperfect data labels, IEEE Trans. Knowl. Data Eng. 26 (7) (2014) 1602–1616.