Research Article

Research on Model and Algorithm of Multiview and Multilabel Classification Based on Nearest-Neighbor Model

Junyi Zhang¹,² and Yuan Rao¹,²

¹Xi’an Key Laboratory of Social Intelligence and Complexity Data Processing, School of Software Engineering, Xi’an Jiaotong University, Xi’an, China
²Shaanxi Joint Key Laboratory for Artifact Intelligence, Xi’an, China

Correspondence should be addressed to Junyi Zhang; zhangjunyi0806@stu.xjtu.edu.cn

Received 20 May 2022; Revised 16 June 2022; Accepted 20 June 2022; Published 22 July 2022

Academic Editor: Lianhui Li

Copyright © 2022 Junyi Zhang and Yuan Rao. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In the field of machine learning, multilabel learning gradually evolved from the traditional text classification problem. In this study, a multiview multilabel via optimal classifier chain (MVMLOCC) algorithm based on the nearest-neighbor model is proposed. The algorithm model establishes a multilabel chain classifier for each view of the dataset and predicts unknown data samples by dynamically adjusting the weights of chain learners. When an unknown example is input, multiple chain classifier models are multiplied by corresponding weights to obtain the final tag set situation. The model makes full use of the relevance between multiple tags and the complementarity and integrity between multiple perspectives and can achieve better learning results. The final experimental results show that our proposed model can be applied to different multilabel classification tasks and has achieved excellent performance under different evaluation indicators. This study mainly studies the multiview and multilabel classification method and uses active learning technology to solve the problem of high label components in data collection.

1. Introduction

Machine learning is the key research direction in the field of artificial intelligence, and its application scope is more and more extensive. In real life, many fields have achieved good results by using machine learning technology [1]. In recent years, with the deepening of scholars’ research, the research scope has expanded from traditional machine learning to deep learning, from binary classification model to multi-classification model, from single-label learning to multilabel learning, from simple single-view learning to multiview learning, and from machine learning to deep learning, and the research direction of machine learning has gradually met the needs of real life [2]. In essence, the classification problem belongs to the sample classification problem, and the traditional classification method mainly aims at learning single label, that is, each sample only needs to be described by one label [3, 4]. However, in practical applications, we will encounter very complex classification problems, namely, multilabel classification, that is, a sample needs to be described by multiple labels. We can see that multilabel classification learning is more universal and more in line with practical applications, which has aroused the research interest of many scholars and become a research hotspot, and has been well applied in different fields such as images, video and audio, and informatics. Therefore, the exploration of multilabel classification learning has a very wide research significance and practical value.

Early research on multilabel learning is to transfer the original mature single-label learning model to multilabel learning. The binary correlation (BR) method is a relatively direct conversion method. Taking classification as an example, it converts multilabel learning problems into multiple single-label binary classification problems [5]. However, its shortcomings are also obvious. This method ignores the dependencies between tags and is prone to produce contradictory prediction results. Compared with
the BR method, the classifier chain can transfer label information among multiple binary classifiers and can use the dependency information between labels to make predictions. Moreover, this method still retains the advantages of simple and easy implementation of the BR method. However, this method also has some shortcomings [6]. In recent years, the study of multiperspective learning has gradually appeared in people's field of vision. People use the diversity brought by different perspectives to put forward multiperspective learning methods, which are widely used and have amazing performance. These views can be obtained from multiple sources or feature subsets.

In order to solve the disadvantage of the single association of traditional multilevel label classification algorithms in the knowledge base, this study will further study the multiview multilabel via optimal classifier chain (MVMLOCC) algorithm based on the nearest-neighbor model [7]. The algorithm can filter a large amount of data in a certain field in a short time, detect and summarize multiple related knowledge point tags of objects in a field, then analyze the features of the detected related knowledge point tags by the feature dimension reduction method, finally analyze the results according to the nearest-neighbor model, construct an association framework of all objects in the field, and complete the multilevel tag classification of knowledge base based on the nearest-neighbor model.

In this study, the formal definition of multiview and multilabel learning problem is given, and a new multilabel learning model is proposed for our multiview combined learning strategy. The innovation contribution is that the model not only combines the consistency of different view data under multiview data, but also combines the correlation between labels in multilabel data, and establishes a chain classifier for the data of each view. At the same time, considering the different prediction effects of each chain, the model gives each chain different weights, which better combines the complementarity and integrity of data from different angles. The experimental results show that the proposed model can be applied to different multilabel classification tasks and has achieved excellent performance under different evaluation indicators.

2. Related Work

2.1. Research Status of Single-Label Classification. The single-label classification is one of the most basic methods in data mining. In real-life applications, people are often faced with complicated things. A single tag can be understood as belonging to a large category, that is, a piece of data is classified and represented by a value. Then, there are two categories and multiple categories within a single label. We can classify some things with similar characteristics together, that is, into the same category by using some rules or standards, and classify very different things into other different categories, which provides convenience for people to analyze and study things, and can predict things more effectively [8, 9].

At present, the classical classification algorithms are naive Bayes classifier (NBC), support-vector machine, association rules, decision tree (DT), K-nearest neighbor, genetic algorithm, neural network, and so on [10–13].

The above classification algorithms have their own characteristics. Based on the practicality and value of these algorithms, they have been widely used in many different fields such as medical diagnosis and treatment, information processing, financial risk assessment, and so on.

2.2. Research Status of Multilabel Classification. In real-life applications, unlike single-label classification, multilabel learning is very common and complex, and it is more in line with the real situation, so it has aroused the research interest of researchers and has become a research hotspot. Up to now, scholars have learned multilabel data in two main ways: the problem transformation method and the algorithm adaptation method.

2.2.1. Problem Transformation Method. The main idea is to transform multilabel learning tasks into other typical tasks. Nowadays, a very common idea is to directly transform multilabel learning tasks into multiple single-label classification tasks. Then, the classical single-label classification algorithm is used to analyze each single-label learning problem, and then, the dataset is artificially classified. Finally, the results of each single-label classification are combined in order to obtain a label vector, and the final classification result is obtained.

2.2.2. Algorithm Adaptation Method. Its idea is to improve the existing classification algorithm, so that it can directly analyze and process multilabel datasets. For example, literature [14] puts forward the ML-KNN algorithm, which combines Bayesian theory with the nearest-neighbor idea to process multilabel datasets, and improves the classical K-nearest neighbor algorithm to get the multilabel classification algorithm ML-KNN, which can effectively solve the multilabel learning problem, and solve the problem of k neighbors, ML-A, and as far as the multilabel learning problem is concerned, a series of studies show that the labels contained in sample instances in the dataset do not exist alone but are related to each other in some form. Literature [15] puts forward the CC algorithm based on the BR classification algorithm, which transforms the multilabel learning task into binary classification functions or classifiers. The main improvement lies in expanding the attribute space of each binary classification function or classifier, that is, adding some column vectors in the attribute space, which are related to labels. Literature [16] puts forward the Bayes chain classifier (BCC), which uses the Bayesian network to learn tags and then obtain the dependency relationship between tags. Although, theoretically, the model or classifier is obtained based on the related information between tags, it also greatly reduces the number of attributes to be extended by each base classifier. However, the disadvantage is that the complexity of constructing a Bayesian network between tags and tag variables will be different from that of tags.

2.3. Challenges Faced by Multilabel Classification Research. In recent years, with the gradual deepening of research, the proposed algorithm has become more efficient and accurate,
and the focus of research has gradually focused on the following points [17–19]:

1. In real life, there are various data sources in different scenarios, so it is more and more difficult to collect complete data, and the obtained data have the problem of incomplete labels. How to better deal with the problem of incomplete labels is the focus of research.

2. In the era of big data, multilabel data are growing exponentially, and labeling data are cumbersome. Manual labeling is not only time-consuming but also has certain deviations. How to label it efficiently has become an urgent problem to be solved. In view of this situation, unlabeled data are easier to obtain and will not be affected by incomplete labeling. How to design an algorithm to actively obtain the information of unlabeled data has become a research direction.

3. There are redundant features in many datasets, such as text and image. These datasets are often high-dimensional and may encounter dimensional disasters in the process of processing. Therefore, it is hoped that the algorithm can solve the problems caused by high-dimensional features and get the importance of features of each dimension in the training process, so as to screen and improve the classification effect of the algorithm.

3. Research Technique

3.1. Nearest-Neighbor Image Selection Method Based on Forward and Backward Filtering. The nearest-neighbor search is to find the most similar items from the database according to the similarity of data. This similarity is usually quantified by the distance between data in space. It can be considered that the closer the data are in space, the higher the similarity between data. Because the nonsemantic nearest-neighbor image contains a large number of noise labels unrelated to the image to be labeled, it is easy to introduce too many noise labels in the label propagation process of the nearest-neighbor image, which reduces the labeling accuracy. Therefore, considering the relationship between the sample to be labeled and the nearest-neighbor image, this study selects the nearest-neighbor sample of the image based on forward and backward filtering and uses it in the tag propagation process based on multi-NMF decomposition.

When calculating the distance between two images, different distance measures are used to calculate the distances on different visual features, and then, these distances are further merged into a global visual distance according to the relative importance of different features. Based on the dynamic distance fusion method [20], the visual distance between image A and image B is defined as follows:

\[
D(A, B) = \frac{1}{R} \sum_{i=0}^{R} \frac{1}{\delta_i^2} d_i(A, B),
\]

(1)
in which \(R\) is the number of visual features, \(d_i(A, B)\) is the distance between image A and image B on the \(i\)th feature, \(\delta_i^2\) is the variance of the distance between all images in the dataset on the \(i\)th feature, \(1/\delta_i^2\) represents the weight of the \(i\)th feature, and smaller \(\delta_i^2\) means that the images in the dataset are relatively close on the \(i\)th feature. On the contrary, it means that there is a big difference.

Due to the uneven distribution of image feature space, some images in the image library have a high probability of being selected as the nearest-neighbor images of other images, while others have a low probability of being selected as the nearest-neighbor images of other images, and some images cannot even be selected as the nearest-neighbor images by any other images, which often leads to the irreversible similarity between the images in the image library and their nearest neighbors.

Therefore, this study proposes a nearest-neighbor image selection method based on forward and backward filtering to select the nearest-neighbor image of the test sample. The so-called forward and backward screening process is the process of bidirectional nearest-neighbor image selection.

First, the image A to be labeled selects its first K visual nearest neighbors and then selects their first K visual nearest neighbors. For a certain nearest-neighbor image B of image A, if its nearest-neighbor image set contains image A, then image B can be regarded as the nearest-neighbor image of image A, that is, image A and image B are each other’s nearest neighbors.

Through the abovementioned positive and negative screening methods, the nearest-neighbor images to be labeled are more in line with the image similarity in the actual situation, and the proportion of the nearest-neighbor images semantically related to the test samples will also increase.

3.2. Label-Specific Feature Transformation. For the multilabel classification algorithm, the algorithm can achieve better classification performance by effectively capturing the unique features of each label. In order to achieve this, it is necessary to study the intrinsic attributes of the labels corresponding to each training instance. There are some labeled data (e.g., training sets and historical test predictions) around the unlabeled data, and their classification results (correct/wrong classification) are known. If the unlabeled instance is close to the misclassified data, it may also be a model error under the local continuity attribute. This property enables us to extract the context properties of unlabeled instances from adjacent tagged data.

For each label \(I_k\), in order to better understand its positive class instance set \(P_k\) and negative class instance set \(N_k\), the multilabel classification algorithm chooses clustering technology, which has been widely used as an independent tool for data analysis.

As suggested in literature [21], based on the simplicity and high efficiency of k-means in clustering analysis, and the better clustering effect when clusters are close to Gaussian distribution, this study also uses the k-means algorithm to cluster the positive and negative instance sets of each label. In other words, for each label, it uses the k-means algorithm
to cluster \( P_k \) and \( N_k \). In order to obtain the unique characteristics of tags, for each tag \( l_k \), \( P_k \) and \( N_k \) are clustered by \( k \)-means, which is used to control the number of clusters. The number of clusters is defined as follows:

\[
a = \lceil \gamma \times \min([P_k], [N_k]) \rceil.
\]

Here, \( \lceil \cdot \rceil \) represents the cardinality of the set, that is, the number of elements in the set \([x]\) represents the largest integer not exceeding \( x \), and \( \gamma \in (0, 1) \) is the scale parameter to control the number of clusters.

For each label, \( l_k \), \( a \) is a fixed value. \( P_k \) is clustered into \( a \) clusters, and the cluster center vector is \((p_{k1}^a, p_{k2}^a, \ldots, p_{ka}^a)\). Similarly, \( N_k \) is clustered into \( a \) clusters, and the cluster center vector is \((n_{k1}^a, n_{k2}^a, \ldots, n_{ka}^a)\). In order to obtain the unique features of labels \( l_k \), for each instance \( x_i \), the feature mapping function \( \varphi_k(x_i) \) is defined as follows:

\[
\varphi_k(x_i) = \{d(x_i, p_{k1}^a), d(x_i, p_{k2}^a), \ldots, d(x_i, p_{ka}^a), d(x_i, n_{k1}^a), d(x_i, n_{k2}^a), \ldots, d(x_i, n_{ka}^a)\}.
\]

Here, the function \( d(\cdot) \) represents the Euclidean distance between the instance and the cluster center.

After all instances are acted upon by the mapping function, we can get the unique feature set \( F_k \) of each tag as follows:

\[
F_k = \{\varphi_k(x_i), y_i(k)\}(x_i, y_i) \in D)\).
\]

Next, for the above description of label-specific feature transformation, pseudocode is used for specific description, as shown in the following Algorithm 1:

3.3. Multiview and Multilabel Optimal Chain Learning Algorithm

3.3.1. Representation and Definition of Multiview and Multilabel Data. The multiclassification problem indicates a classification problem with more than 2 categories. Multiclassification problem is to add categories on the basis of two classification problems. Such problems are based on the assumption that a sample belongs to only one category. Multilabel problem is as follows: the purpose of this kind of problem is to add a series of target labels to each sample. The important difference between this problem and multiclassification problem is that each tag is not mutually exclusive. Here, we first explain the formal definition of multiview and multilabel classification. The following definitions are applied to the whole study. In the multiview and multilabel classification problem, a given dataset contains \( n \) data samples, each of which has a subset of \( L \) category labels, which come from \( M \) different independent perspectives, and for each perspective, there will be an output space of \( n \) labels.

We use matrix \( [(X^m_{1}, y^m_{1}), \ldots, (X^m_{n}, y^m_{n})], X^m \in R^{dm}, i \in \{1, 2, \ldots, n\}, m \in \{1, 2, \ldots, M\} \) to represent the input dataset and \( y^m_{nl} \in \{-1, +1\}^L \) to represent the \( L \) possible labels of the output, where \( y^m_{nl} \) indicates that the sample corresponds to the \( l \)-th label, and if it is \(-1\), it means that the sample is irrelevant to the label. The following chapters are all expanded in the data form of this section.

3.3.2. Chain Classifier Algorithm. Many efficient models have been put forward for multilabel classification task after an in-depth study by scholars, among which the binary relevance (BR) algorithm is a typical method under the problem transformation strategy. This algorithm model directly decomposes multilabel classification into several independent single-label classification problems. First, the dataset is decomposed to get the dataset for each label, and then, the classification learners corresponding to the labels are established, respectively.

Subsequently, the classifier chain method proposed by scholars was established on the basis of the BR method combined with the relevance of tags, and the classification effect of this model was improved to some extent [22, 23]. Classifier chain (CC), as a problem transformation strategy, is solved by designing a chain composed of multiple binary classifiers. In essence, the CC algorithm is an improvement of the BR method. When an unknown sample is input into the algorithm, the model first predicts the condition of the first label and then takes this sample example together with the predicted value of its corresponding label as the training set sample for predicting the next label. By analogy, the condition of each label will be obtained in turn, and the final output result will be obtained by synthesizing the results of each classifier.

3.3.3. Multilabel and Multiview Optimal Chain Algorithm. The multilabel and multiview optimal chain algorithm originated from the abovementioned chain classifier idea. The labels around the classifier created each time in the classifier chain are randomly selected, which may bring prediction errors. In the field of machine learning, higher accuracy is always the theme of constant pursuit. To further improve the classification accuracy, we propose a multiview multilabel via optimal classifier chain (MVMLOCC) model to learn a set of classifier chains, and each chain classifier represents a learner learned from a corresponding perspective.

For the sake of simplicity, this study only introduces how to generate a chained multilabel classifier from multiple perspectives. Every \( M \) data example can be represented by feature sets of \( M \) perspectives. Each perspective corresponds to a representation form of data, and we train and learn an optimal chain classifier for each perspective, in which each base classifier is composed of a support-vector machine (SVM) [24]. At the same time, in order to verify the better effect of multiview learning, we splice the feature sets obtained from multiple views into a larger feature set and then
build an optimal chain model based on this feature set for classification prediction.

In order to give a clearer representation of the model, the following forms are given:

\[ H(x) = \sum_{m=1}^{M} \alpha_m h_m(x). \]  

(5)

The basic function \( h_m(x) \) in formula (5) is the output condition of each label of the \( m \)-th classifier chain for the position sample. \( \alpha_m (i) \) is a weight factor, which represents a measuring factor for the output prediction result of the \( l \)-th label in the \( m \)-th classifier chain.

The classifier chain model trains and learns a binary classifier for each label according to the order of randomly selected labels. This order is randomly selected, which may bring poor prediction accuracy. We improve the performance of the model by designing an ordered sequence of labels sorted according to their relevance. Here, we define a weight network graph \( G = (V, E, W) \) to describe the complex correlation, where \( V \) represents the node set (i.e., label set) of the network, \( E \) represents the connected edge of any two labels, and \( W \) represents any two label association weights (i.e., the correlation between labels) set.

Each category label is represented by a label vector, so the correlation between the two labels can be calculated by cosine similarity, that is, any two labels (i.e., two row vectors) are selected from the matrix to calculate cosine similarity, and the smaller the calculated value, the stronger the correlation between the two labels. Here, we use an \( L \times L \) matrix \( B \) to represent the correlation between two tags. Each element \( b_{ij} \) in the matrix represents the correlation between the corresponding \( A_i \) tag and \( A_j \) tag. The calculation formula (6) is as follows:

\[ b_{ij} = \text{sim}(A_i, A_j) = \cos(\Theta_{ij}) = \frac{\sum_{m=1}^{N} a_{m,i} a_{m,j}}{\sqrt{\sum_{m=1}^{N} a_{m,i}^2 \sum_{m=1}^{N} a_{m,j}^2}}. \]  

(6)

Therein, \( a_{i}^{m} \) represents the value in the \( i \)-th row and \( n \)-th column of the association matrix \( A \), that is, the \( n \)-th component element in the vector of the \( i \)-th label. Here, we calculate the spatial distance between two row vectors to get the correlation between two labels.

Through formula (6), we can calculate the correlation between any two tags. Furthermore, we can get the network structure diagram \( G = (V, E, W) \) of tag correlation, in which the weight value of \( W \) is the correlation between any two tags calculated by us.

In the model, we use the training set to learn from \( M \) different perspectives to obtain \( M \) optimal classifier chains. Because different samples have different labels, the different correlations between the labels will affect the final model prediction effect. Here, we assign the corresponding weight value \( \alpha_m \) to the prediction effect of \( M \) classifier chains.

\[ Q(\alpha) = \sum_{m=1}^{M} \frac{1}{2} \left\{ \alpha_m \| \hat{Y} - Y \|_F^2 \right\} s.t \sum_{m=1}^{M} \alpha_m^2 = 1, \alpha \geq 0. \]  

(7)

The idea of determining the weight value of the multilabel classifier chain by training error is obtained from this study [25]. Here, we set \( r > 1 \) to avoid that the model only considers the output result of a single classifier chain due to a trivial solution, thus ensuring the complementarity of multiple classifier chains.

Algorithm 2 shows the specific implementation algorithm of the MVMLOCC model. After we get the weight vector, we can use formula (5) to predict the unknown samples and output the results.

### 4. Experimental Research

#### 4.1. Experimental Setup

4.1.1. Dataset. In this experiment, two open multiview multilabel datasets (Corel5K, ESP game, and PASCAL VOC) are used. The data come from many different fields. Each sample in the dataset is a multilabel image, and each image has a corresponding label subset based on its content. In the experiment, we use the standard training/test partition ratio, and the total number of labels in each dataset is different. The specific statistical information of the data used in the test is shown in Table 1. All the data are from Mulan, and more detailed information can be found here.

4.1.2. Contrast Algorithm. In order to test the performance of our method in multiclassification and multilabel classification, we will compare it with the multiview algorithm of multifeature level fusion and classifier-level fusion. The related comparison algorithm will be introduced first.
In order to select the appropriate number of nearest-neighbor images, this study makes corresponding verification experiments under the condition that the number of nearest-neighbor images is different. During the experiment, the weight parameters of the image visual feature angle are all set to 0.01, the weight parameters of the tag feature angle are set to 1, the number of tag allocation is set to 5, and the number of potential topics is set to 150, as shown in Figure 1.

With the increase in the number $K$ of the nearest-neighbor images of the target image, the performance indexes of the image annotation method based on multi-NMF decomposition generally show a trend of increasing at first and then tending to be stable. Taking the curve of average recall obtained from the experiment in the Corel 5K database as an example, when the value of $K$ is less than 50, the value of average recall gradually increases with the increasing number of nearest-neighbor images $K$.

When the value of $K$ is greater than 50, with the value of $K$ increasing, the value of average recall has not significantly increased but has remained relatively stable. Therefore, when the number $K$ of nearest-neighbor images is not large, the value of the average recall rate will remain stable. Therefore, due to the compromise between labeling accuracy and processing time, the number $K$ of nearest-neighbor images is generally set to 50.

### 4.2.2. Performance Comparison

In the MVMLOCC model, we set up an optimal classifier chain for data subsets from each perspective. In the experiment, we selected two different perspectives from the data for the experiment.

The data in Table 2–Table 6 are the prediction effect evaluation of each algorithm in two datasets under different evaluation standards. The specific sample numbers of the training set and the test set used in the experiment are randomly divided.

Based on the above experimental results, we can observe the following:

1. Compared with other methods, the performance of our proposed algorithm model on two datasets has certain advantages, which proves the efficiency of our method.
2. As mentioned earlier, the BR method assumes that labels are independent of each other, so each category label is treated as a separate classification problem, and the performance advantage obtained in the experiment is not obvious.
3. Compared with the hierarchical SVM multilabel classification algorithm, our algorithm model performs slightly worse on a given dataset, but our method shows better stability when all datasets are

### Algorithm 2: Multiview and multilabel optimal chain classifier learning algorithm.

1. **SVM.** We use standard SVM to deal with every single perspective data and all data from different perspectives. In other words, we use each single perspective feature to train different SVM, and after all features are spliced together, we train to be a SVM.

2. **GP-PMK.** GP-PMK adopts the Gaussian process method and pyramid matching kernel function to improve the classification effect.

3. **ML-KNN.** Multilabel nearest-neighbor (ML-KNN) is proposed to solve the problem of multilabel classification from a single perspective. The ML-KNN algorithm obtains the statistical information from the neighbor samples of unknown samples and determines which categories the unknown samples belong to by using the method of maximum a posteriori probability.

4. **Hierarchical SVM.** The algorithm belongs to the multiview algorithm of classifier-level fusion. First, each view feature is used to train an independent SVM, and then, the prediction result is used as the input of another SVM. In our experiment, we realize hierarchical SVM by using standard SVM.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Corel5K</th>
<th>ESP game</th>
<th>PASCAL VOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total samples</td>
<td>4999</td>
<td>20770</td>
<td>10000</td>
</tr>
<tr>
<td>Number of training samples</td>
<td>4500</td>
<td>18689</td>
<td>5011</td>
</tr>
<tr>
<td>Number of test samples</td>
<td>499</td>
<td>2081</td>
<td>4952</td>
</tr>
<tr>
<td>Total tags</td>
<td>266</td>
<td>268</td>
<td>20</td>
</tr>
</tbody>
</table>
integrated. To sum up, the experimental results can prove the efficiency of our method.

4.2.3. Runtime Analysis and Convergence Analysis. In order to evaluate the performance of the MVMLOCC algorithm, we select some comparison algorithms to compare their running time. The comparison algorithms include linear SVM using all features, GP-PMK using all features, ML-KNN, and hierarchical SVM. In Figure 2, we show the running time of different algorithms based on a different number of samples. As shown in Figure 2, the methods based on SVM and MVMLOCC combined with different norm constraints are all nonlinear methods, while our method has linear computational time complexity. When dealing with large-scale multiview data, the difference in running time between nonlinear and linear methods will be particularly large. In addition, it is worth noting that the parallel version of MVMLOCChas a lower running time on different data than linear SVM and ML-KNN methods, which further proves that our method is suitable for processing large-scale multiview data.

Table 2: Ranking loss performance comparison.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>SVM</th>
<th>GP-PMK</th>
<th>ML-KNN</th>
<th>Hierarchical SVM</th>
<th>MVMLOCC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Best SV</td>
<td>Con SV</td>
<td>Best SV</td>
<td>Con SV</td>
<td>Best SV</td>
</tr>
<tr>
<td>Corel5K</td>
<td>0.936</td>
<td>0.205</td>
<td>0.236</td>
<td>0.205</td>
<td>0.125</td>
</tr>
<tr>
<td>ESP game</td>
<td>0.114</td>
<td>0.244</td>
<td>0.337</td>
<td>0.217</td>
<td>0.102</td>
</tr>
<tr>
<td>PASCAL VOC</td>
<td>0.201</td>
<td>0.336</td>
<td>0.174</td>
<td>0.225</td>
<td>0.107</td>
</tr>
</tbody>
</table>

Table 3: Average precision performance comparison.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>SVM</th>
<th>GP-PMK</th>
<th>ML-KNN</th>
<th>Hierarchical SVM</th>
<th>MVMLOCC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Best SV</td>
<td>Con SV</td>
<td>Best SV</td>
<td>Con SV</td>
<td>Best SV</td>
</tr>
<tr>
<td>Corel5K</td>
<td>0.305</td>
<td>0.302</td>
<td>0.487</td>
<td>0.325</td>
<td>0.225</td>
</tr>
<tr>
<td>ESP game</td>
<td>0.647</td>
<td>0.335</td>
<td>0.776</td>
<td>0.741</td>
<td>0.617</td>
</tr>
<tr>
<td>PASCAL VOC</td>
<td>0.120</td>
<td>0.207</td>
<td>0.174</td>
<td>0.281</td>
<td>0.217</td>
</tr>
</tbody>
</table>

Table 4: Hamming loss performance comparison.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>SVM</th>
<th>GP-PMK</th>
<th>ML-KNN</th>
<th>Hierarchical SVM</th>
<th>MVMLOCC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Best SV</td>
<td>Con SV</td>
<td>Best SV</td>
<td>Con SV</td>
<td>Best SV</td>
</tr>
<tr>
<td>Corel5K</td>
<td>0.063</td>
<td>0.066</td>
<td>0.225</td>
<td>0.147</td>
<td>0.001</td>
</tr>
<tr>
<td>ESP game</td>
<td>0.147</td>
<td>0.125</td>
<td>0.417</td>
<td>0.263</td>
<td>0.201</td>
</tr>
<tr>
<td>PASCAL VOC</td>
<td>0.301</td>
<td>0.147</td>
<td>0.228</td>
<td>0.247</td>
<td>0.144</td>
</tr>
</tbody>
</table>

Table 5: Comparison of one-error performance.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>SVM</th>
<th>GP-PMK</th>
<th>ML-KNN</th>
<th>Hierarchical SVM</th>
<th>MVMLOCC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Best SV</td>
<td>Con SV</td>
<td>Best SV</td>
<td>Con SV</td>
<td>Best SV</td>
</tr>
<tr>
<td>Corel5K</td>
<td>0.174</td>
<td>0.225</td>
<td>0.258</td>
<td>0.201</td>
<td>0.201</td>
</tr>
<tr>
<td>ESP game</td>
<td>0.136</td>
<td>0.174</td>
<td>0.206</td>
<td>0.307</td>
<td>0.013</td>
</tr>
<tr>
<td>PASCAL VOC</td>
<td>0.217</td>
<td>0.228</td>
<td>0.146</td>
<td>0.528</td>
<td>0.392</td>
</tr>
</tbody>
</table>

Table 6: Comparison of coverage performance.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>SVM</th>
<th>GP-PMK</th>
<th>ML-KNN</th>
<th>Hierarchical SVM</th>
<th>MVMLOCC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Best SV</td>
<td>Con SV</td>
<td>Best SV</td>
<td>Con SV</td>
<td>Best SV</td>
</tr>
<tr>
<td>Corel5K</td>
<td>0.302</td>
<td>0.307</td>
<td>0.471</td>
<td>0.336</td>
<td>0.021</td>
</tr>
<tr>
<td>ESP game</td>
<td>0.335</td>
<td>0.322</td>
<td>0.258</td>
<td>0.205</td>
<td>0.217</td>
</tr>
<tr>
<td>PASCAL VOC</td>
<td>0.014</td>
<td>0.724</td>
<td>0.218</td>
<td>0.258</td>
<td>0.326</td>
</tr>
</tbody>
</table>
Figure 1: Labeling performance curves with a different number of nearest-neighbor images.
5. Conclusions

This model not only combines the consistency of different view data under multiview data, but also combines the correlation between labels in multilabel data, and establishes a chain classifier for each view data. Considering the different prediction effects of each chain, the model gives each chain different weights, which better combines the complementarity and integrity of data from different angles. When dealing with large-scale multiview data, the running time difference between nonlinear methods and linear methods will be particularly large. In addition, it is worth noting that compared with linear SVM and ML-KNN methods, the parallel version of MVMLOCC has a shorter running time on different data, which further proves that our method is suitable for processing large-scale multiview data. Experiments show that the performance of the multiview and multilabel learning method is far better than that of the traditional multilabel algorithm. This method provides a new strategy for solving multilabel classification problems from multiple perspectives [26].

Data Availability

The experimental data used to support the findings of this study are available from the corresponding author upon request.
Conflicts of Interest

The authors declare that they have no conflicts of interest regarding this work.

Acknowledgments

The research work was supported by the National Key Research and Development Program in China (2019YFB2102300); the World-Class Universities (Disciplines) and the Characteristic Development Guidance Funds for the Central Universities (PY3A022); the Ministry of Education Fund Projects (18)ZD022 and 2017B00030); the Shenzhen Science and Technology Project (JCYJ20180306170836595); Basic Scientific Research Operating Expenses of Central Universities (No. ZDYF2017006); the Xi'an Navinfo Corp. & Engineering Center of Xi'an Intelligence Spatial-Temporal Data Analysis Project (C2020103); and the Beilin District of Xi'an Science & Technology Project (GX1803).

References