A Hierarchical Algorithm for Image Multi-labeling

Jiwei Hu\textsuperscript{1}, Kin Man Lam\textsuperscript{1}, Guoping Qiu\textsuperscript{2}

\textsuperscript{1}. Centre for Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong
\textsuperscript{2}. Department of Computer Science, Nottingham University, UK

ABSTRACT

This paper presents an efficient two-stage method for multi-class image labeling. We first propose a simple label-filtering algorithm (LFA), which can remove most of the irrelevant labels for a query image while the potential labels are maintained. With a small population of potential labels left, we then apply the Naive-Bayes Nearest-Neighbor (NBNN) classifier as the second stage of our algorithm to identify the labels for the query image. This approach has been evaluated on the Corel database, and compared to existing algorithms. Experiment results show that our proposed algorithm can achieve a promising result, as it outperforms existing algorithms.

Index Terms—Multi-label classification, Label filtering, Nearest Neighbors

1. INTRODUCTION

Nowadays, with the increasing development of digital multimedia and computer technology, image archives are growing at a remarkable rate. Image annotation has become an increasingly important and popular research topic; yet it is also a difficult task. The annotation process can be considered a multi-label classification, while the traditional classification method usually associates each image with a single label class. In the real world, however, an image is usually associated with more than one label, which is characterized by different regions in images. Thus, image annotation is naturally posed as a multi-classification problem rather than a single-class classification problem. The existing methods for multi-label classification can be grouped into two categories. One is called problem transformation methods \cite{6}, which transform the multi-label classification problem into one or several single-label classification problems. The other is called algorithm adaptation methods \cite{2}, which use specific algorithms to handle multi-label data directly. The first approach requires an intensive learning stage, so it is a supervised learning approach. The second approach usually sacrifices accuracy for the sake of efficiency.

AdaBoost.MH and AdaBoost.MR \cite{8}] employ the popular AdaBoost \cite{7} method to solve multi-label classification problems. In Adaboost.MH, for a new sample $x$ and a label $l$, if the output of the weak classifier, which is a simple classifier based on a single feature, is positive, we consider that the sample $x$ can be labeled as $l$. Otherwise, $l$ is not a label of the sample $x$. In contrast, for Adaboost.MR, the output of the weak classifier is used for ranking all the potential labels.

Zhou and Zhang \cite{4} presented a multi-label lazy learning approach, named ML-$k$NN, derived from the traditional $k$-NN algorithm. For each single label $l$, the ML-$k$NN finds the $k$ nearest neighbors (NN) to the test image and considers those containing the label $l$ as positive, and the rest as negative.

Our proposed method aims to remove most of the irrelevant labels in the first step. In real situations, there are a large number of photos accompanied by an even larger number of labels. To design a practical image-annotation system, we have to consider both the efficiency and the accuracy of the approach to be used. The aim of this paper is to design a simple method which can remove most of the irrelevant labels efficiently while keeping the potential labels. Our method is similar to the label-transfer method proposed by Makadia et al. \cite{3}. However, our method as a whole is quite different to \cite{3}, which is improved so that it is suitable for multi-label classification.

Having removed a large portion of the labels available that are irrelevant to a query image, the method proposed by Oren et al. \cite{1} will be extended and employed in the second stage of our algorithm. Oren’s method employs the NBNN, which uses a distance metric in the space of the local image descriptors. NBNN computes the direct ‘Image-to-Class’ distances without quantizing the descriptors used. Our algorithm uses the concept of NBNN, and we improve it so that it is suitable for our multi-label classification problem.

This paper is organized as follows. Section 2 discusses our proposed label-filtering algorithm (LFA), which is the algorithm used in the first stage of our method to remove irrelevant labels. Section 3 will present the details of our improved version of the NBNN method. In Section 4, we describe the setup of our experiments and the results. Section 5 provides the conclusion.
2. LABEL FILTERING ALGORITHM

Our label-filtering algorithm (LFA) is motivated by the label-transfer method [3] proposed by Makadia et al. However, our algorithm is quite different from their label-transfer method in the sense that we make use of statistical data and the prior knowledge of the dataset, and also consider the relationships among the labels. Our algorithm aims to filter most of the irrelevant labels efficiently, and hence reduce the number of potential labels for the testing images.

Suppose that \( n \) potential labels are to be filtered to represent a query image \( I \). These potential labels are obtained from the labels associated with the \( K \) nearest neighbors of the image \( I \) using the \( k \)-NN search. The label-filtering algorithm is described as follows:

1. Suppose the \( K \) nearest neighbors have \( q \) different labels. Count the respective frequencies of the \( q \) labels based on the \( K \) nearest neighbors, and sort the labels into descending order of frequency. These sorted labels are denoted as \( f_i \), \( i = 1, \ldots, q \).
2. Compute the average \( \alpha \) of the frequencies of all the labels.
   \[
   \alpha = \frac{1}{q} \sum_{i=1}^{q} f_i.
   \]
3. Choose the first \( l \) labels, with their frequencies being higher than \( \alpha \). These \( l \) labels are considered the potential and relevant labels of the query image \( I \).
4. Denote \( p(x,y) \) as the co-occurrence matrix of the samples \( x \) and \( y \). Calculate the co-occurrence matrix of the remaining \( q-l \) labels and the \( l \) labels chosen in Step 3. Then, compute the sum of each column to obtain the value \( C_i \), where \( i = 1, 2, \ldots, n-l \).
5. Denote \( F_j \) as the frequency of occurrence of label \( j \) in the training set. Calculate the normalized frequency \( F_j^* \) of the \( n-l \) labels in the training set as follows:
   \[
   F_j^* = F_j / \sum_{i=1}^{q-l} F_i, \text{ where } j = 1, 2, \ldots, n-l.
   \]
6. Compute the product of \( C_i \) and \( F_j^* \), which represents the degree of likelihood of the \( j \)-th label being a label of the query image. These sums are then normalized as follows to form a probability distribution \( p_i \) (\( i = 1, 2, \ldots, q-l \)) for the \( q-l \) labels.
   \[
   p_j = F_j^* C_j / \sum_{i=1}^{n-l} (F_i^* C_i), \text{ where } j = 1, 2, \ldots, n-l.
   \]
Finally, a threshold value \( T_1 \) will be set such that those labels with probabilities higher than the threshold, i.e. \( p_i > T_1 \), will be selected as labels for the query image. In other words, those labels with their \( p_i \) smaller than \( T_1 \) will be regarded as irrelevant. A maximum number of labels for an image can also be set.

The filtering algorithm is simple and efficient. At this stage, we are not aiming to achieve accurate labeling, but rather to achieve a missing rate that is as low as possible. The issue of overall accuracy is dealt with in the second stage of our algorithm.

3. PATCH-BASED NBNN CLASSIFIER

Traditional image classification methods always need an intensive training stage with a lengthy training time. Conversely, the non-parametric nearest-neighbor-based image classifiers do not require a training stage, but this is very computationally intensive if many training samples are available.

Oren et al. argued that quantization always gives rise to significant degradation in the discriminative power of descriptors, although the dimension can be greatly reduced. They also claimed that the simple non-parametric algorithms that do not require a training phase can “undo” damage caused by quantization. A simple and efficient method, namely NBNN, was proposed. Following is a brief description of the NBNN classifier.

Given a query image, compute all its local image descriptors \( d_1, \ldots, d_n \). The class \( C \) with the sum \( \sum_{i=1}^{n} ||d_i - NN_c(d_i)|| \) minimized is searched, where \( NN_c(d_i) \) is the most similar descriptors of \( d_i \) in class \( C \).

The original NBNN method was proposed to solve the binary classification problem, so we extend it to multi-label classification. In our method, we divide each image into 4 patches. Each patch is considered an interest part of the image under consideration, and the features of each part are extracted. The \( k \)-means algorithm is then used to cluster all the patch features in the training set in order to generate a codebook for each class. The codebook size can be adjusted so as to control the local information of the images to be maintained. With the codebooks constructed for the label classes, the patch features of a query image are extracted and compared to the corresponding patch features in each codebook. The minimum distance for each of the 4 patches is computed, and the total distance is computed by summing up these four minimum distances. Then, these distances are sorted into ascending order, and the first \( N_k \) of the labels are assigned to the query image.

4. EXPERIMENTS

The experiments are divided into two parts. First, we will evaluate the effectiveness of the LFA algorithm in removing irrelevant labels for testing samples. Second, we will measure the performances of our two-stage framework for image annotation.

The database used is Corel5K, which has been widely adopted as an evaluation benchmark for image annotation. It contains 5,000 images: comprising 4,500 training and 500 testing samples. Each image in the dataset is annotated with
about 3.5 keywords on average, and the dictionary has a total of 374 words or labels.

The aim of the first stage of our algorithm is to efficiently remove most of the irrelevant labels from the testing images. The features to be used have a vital effect on the performance. We have tested the performances of different features, such as RGB, Haar, Gabor, etc. Experiments show that the features based on the Colored Pattern Appearance Model (CPAM) [11] are simple and efficient.

4.1 Label Filtering

As mentioned previously, the total number of labels available for the training images is 374. However, the 500 testing images cover only some of the labels. Thus, we selected 4007 from 4500 training images with 110 labels only, which cover all the labels of the testing images.

In the first step, $K$ is set at 100, i.e. we select the 100 training samples which are nearest to the query image. The labels of these 100 images may be the same, so the number of distinct labels obtained ranges from 30 to 60, with the 500 testing images.

In the experiment, three parameters are considered. First, the co-occurrence between the $l$ labels defined by Step 2 and the other $n-l$ potential labels. The value of $l$ used will have a significant effect on the performance of our algorithm because these $l$ labels will be considered as relevant labels. We choose the $l$ labels in our algorithm whose frequencies are higher than $2.4\sqrt{n}$ (we set $\alpha = 2$ by experiment) as described in Step 3 of our algorithm. Then, the respective local frequencies of these $q-l$ labels are computed. After normalization, two probability matrices are obtained. The dimension of the co-occurrence matrix $p(x,y)$ is $m \times n$, where $m$ is the number of keywords of the $l$ labels, and $n$ is the number of keywords of the $q-l$ potential labels. The dimension of the local frequency matrix $F_x$ is $1 \times n$. The product of the above two matrices, i.e. $p(x, y)F_x$, provides the probabilities of the $q-l$ labels. Actually, we have taken the frequencies of the first $l$ labels into consideration; however, the results do not change much. We can therefore conclude that the co-occurrence between the $l$ labels and the other $q-l$ labels is a more important factor for our algorithm than the corresponding frequencies of the $l$ labels as appeared in the training set. The performance of our proposed method is compared to the label-transfer method by Makadia et al., as shown in Fig. 1.

Fig. 1 shows that our label-filtering algorithm outperforms the label-transfer method in removing irrelevant labels. Experiment results show that our method can efficiently remove most of the irrelevant labels. The average number of labels we chose as our potential labels is 4.1. We conclude that the co-occurrence between the frequencies of the first $l$ labels and the other $q-l$ labels is a more important factor for our algorithm than the frequencies of the $l$ labels as appeared in the training set. The performance of our proposed method is compared to the label-transfer method by Makadia et al., as shown in Fig. 1.

4.2 Combined with a distance metric

In order to compare our method with the label-transfer method, we also follow the experiment by Makadia et al. We employ the label-transfer method and the proposed label-filtering algorithm to their baseline annotation methods, which comprise a composite image distance measure (JEC and Lasso). We evaluate the two methods based on the Corel5K database.

We choose three of the five measurements described in [3] to evaluate the performances of the label-transfer method and the label-filtering method, i.e. the mean precision rates ($P\%$), the mean recall rates ($R\%$), and the number of total keywords recalled ($N^*$). The experiment results shown in Table 1 prove that the proposed method achieve a better performance for image annotation.

4.3 Image annotation

In this part of the experiments, we evaluate the performances of our whole framework, i.e. the two-stage algorithm, for image annotation. As mentioned in Section3, we employ the patch-based NBNN classifier to annotate the testing images.
The features to be used will affect the performances of the image annotation algorithm. Similar to [5], we use different feature descriptors, and also a combination of these features in our experiments. The following features are used:

1. Color feature: RGB color moment (3×3 grid, color mean, variance, skewness for R,G,B),
2. Edge histogram (edge orientation histogram, Canny edge detector),
3. Gabor wavelets transform (5 scales and 8 orientations, 3 moments for each sub-image),
4. Local Binary Pattern (59-dimensional LBP histogram),
5. GIST (complex and popular feature descriptor).

We compare our proposed framework with the label-propagation algorithm in [7] and with an efficient image-annotation approach proposed in [10]. [7] applies a label-propagation algorithm to assign the label posterior probability to images using information from available unannotated images in a semi-supervised manner. [10] proposes a novel approach to image annotation based on a semantic distance function (SDF). This approach can simultaneously learn a semantic distance by capturing the prior annotation knowledge and propagate the annotation of an image as a whole entity.

As can been seen from Fig. 2, our method’s performance is a marked improvement on the traditional label-propagation method. Our method also outperforms a very efficient annotation method that uses the label-propagation idea. Our algorithm combines the advantages of the label-propagation idea and the efficient NBNN classifier to effectively achieve a very promising result.

**5 CONCLUSIONS**

This paper presents an efficient hierarchical image annotation system which employs the ideal of label propagation to the NBNN classifier. This proposed framework can efficiently remove most of the irrelevant labels, and it solves the multi-label problem faced by images belonging to a number of classes. The hierarchical framework proposed in our algorithm can efficiently remove most of the irrelevant labels from testing images, and then a more computational, but also more accurate method is employed for image annotation. This architecture can therefore achieve both efficiency and accuracy. In our experiment, our algorithm can achieve the accuracy rates of 94% of the testing images having all their labels annotated, and 97% of them having at least one label annotated.

**Acknowledgement:** The work described in this paper was fully supported by a grant from the Research Grants Council of the HKSAR, China ((Project No. PolyU 5192/07E).

**6. REFERENCES**