Locating High-density Clusters with Noisy Queries

Chen Cao[1], Shifeng Chen[1], Changqing Zou[1], Jianzhuang Liu[1,2]
[1]*Shenzhen Key Laboratory for Computer Vision and Pattern Recognition*
Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China
[2]*Department of Information Engineering, The Chinese University of Hong Kong, China*
{*chen.cao, shifeng.chen, cq.zou, jz.liu*}*@siat.ac.cn*

Abstract

Semi-supervised learning (SSL) relies on a few labeled samples to explore data's intrinsic structure through pairwise smooth transduction. The performance of SSL mainly depends on two folds: (1) the accuracy of labeled queries, (2) the integrity of manifolds in data distribution. Both of these qualities would be poor in real applications as data often consist of several irrelevant clusters and discrete noise. In this paper we propose a novel framework to simultaneously remove discrete noise and locate the high-density clusters. Experiments demonstrate that our algorithm is quite effective to solve several problems such as non-feedback image re-ranking and image co-segmentation.

1. Introduction

Semi-supervised learning (SSL) methods regard data as nodes to construct pairwise edges between them. These methods hold a key assumption that nearby nodes have similar labels, therefore nodes in a same semantic group are likely to form a cluster. Given a few labeled data as queries, the entire unlabeled data could be classified based on their similarities to queries. The similarity is measured through edge transduction.

The solution to SSL problem is a minimization on both data smoothness and empirical risk. The smoothness is measured by data's intrinsic structure, and the empirical risk depends on initial queries. This model is widely used in applications such as image retrieval [12] and interactive graph cut [1]. In real world, data manifolds would be destroyed by noise, which lead to deviation of smooth transduction. Furthermore, when the initial queries are generated automatically instead of manual labeling, the inaccurate labels would finally bring about unsatisfactory classification results.

Here we focus on such a scenario: a few inaccurate positive labels are used to find interesting data clusters with high density. The scenario is general in many applications, *e.g.*, (1) web image re-ranking when keywords-based top-ranked images are used as positive queries; (2) image co-segmentation when users roughly stroke on intended foreground. Both the targets (intended images and foreground) are likely to form high-density clusters, while irrelevant data would be small clusters and scattered noise. Our algorithm aims at removing noise and finding data in the high-density clusters, by given a few imprecise positive samples.

To illustrate this learning issue, we introduce a toy data composed of manifolds and outliers in Figure 1. Our goal is to extract data in the high-density cluster by given a few inaccurate labels. Two recent noise resistant *SSL* methods LabelDiag [10] and SpecFilter [7] make efforts to handle this kind of problem by devising filters to purify the noisy labels. We compare our algorithm with these methods and the standard *SSL* approach (*LGC*) [13] which directly uses noisy labels.

The main difference between our method and recent approaches [10, 7] lies in two folds: (1) we consider global discrete noise removal, while [10, 7] only filter noise in initial queries; (2) we extract accurate queries from the highest density cluster. LabelDiag [10] removes a negative sample and simultaneously adds a positive sample to query set at each step, which would bring more noise when the precision of initial query set is low. SpecFilter [7] locates local dense multi-regions in query set, which would lead to entire wrong labels when noise form small compact clusters.

2. Algorithm Description

Semi-supervised learning starts from constructing graph on dataset. The n-element dataset is regarded as nodes set $\mathcal{X} = \{x_1, \cdots, x_l, x_{l+1}, \cdots, x_n\}$. Edges are built between every two nodes to form an $n \times n$ weight matrix W as $w_{ij} = \exp\left(-\|x_i - x_j\|^2 / 2\sigma^2\right)$ if $i \neq j$ and $w_{ii} = 0$. W is also normalized as

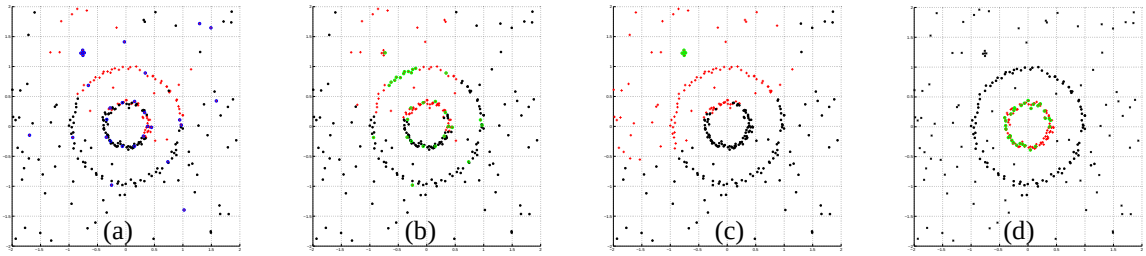

Figure 1. Experiments on two-circle toy data. (a) *LGC* [13] using initial noisy queries. (b) Label diagnosis [10]. (c) Spectral filter [7]. (d) Our method. The goal is to extract the inner circle (112 points) via noisy queries. Blue markers in (a) indicate initial queries, green markers in (b)(c)(d) indicate purified queries by each method, red markers indicate the experimental results (top 112 ranked points) of each method, and the "cross" markers in (d) indicate globally removed noise by our method. The figure is best viewed by magnifying 600%.

$S = D^{-1/2}WD^{-1/2}$, where D is a diagonal matrix with $d_{ii} = \sum_{j=1}^{n} w_{ij}$. We define a query vector $\mathbf{y}$ with $y_i = 1$ (labeled "positive") and $y_u = 0$ (unlabeled). In our application, we suppose that the first l elements in $\mathcal{X}$ are initialized as "positive", but the labels are inaccurate. Our goal is to refine $\mathbf{y}$ to improve the accuracy.

2.1. Regularization model

Given an l-node query set $\mathcal{X}_l = \{x_1, x_2, \cdots, x_l\}$ and an $l \times 1$ query vector $\mathbf{y}_l = \mathbf{1}$ in all dimensions, our goal is to learn a discriminant function $f(\cdot)$ so that intended/positive nodes would have relative larger values. Here we assume $f(\cdot)$ to be linear as

$$f(x; \mathbf{w}) = \langle \mathbf{w}, \Psi(x) \rangle , \tag{1}$$

where $\mathbf{w}$ denotes a weight parameter which has the same dimension as feature vector $\Psi(x)$. In Section 2.2, we will discuss how to choose representative $\Psi(\cdot)$.

Ideally, $f(x_i; \mathbf{w}) = 1$ when x_i is positive. We use the initial labeled "positive" data to build the model

$$\mathbb{Q} = \frac{1}{2}\left(\sum_{i=1}^{l} \| f(x_i; \mathbf{w}) - 1 \|^2 + \mu \langle \mathbf{h}, \mathbf{w} \bullet \mathbf{w} \rangle \right) , \tag{2}$$

where $\mathbf{h}$ is to assign different weights for different dimensions of $\Psi(x)$, and $\bullet$ is point-multiplication operator. In the brackets, the first term is empirical risk to keep $f(x; \mathbf{w})$ not change too much from initial labeled queries, and the second term is a regularizer to restrict small $\mathbf{w}$ and avoid over-fitting issue. μ is a balance factor between the two terms. By minimizing $\mathbb{Q}$ with $\mathbf{w}$, x_i would get relative larger $f(x_i; \mathbf{w})$ if $\Psi(x_i)$ has lots of similar peers in query set. Conversely, x_j with singular $\Psi(x_j)$ would get relative smaller $f(x_j; \mathbf{w})$.

Here we define $X = [\Psi(x_1), \cdots, \Psi(x_l)]^T$ and a diagonal matrix $H = diag(\mathbf{h})$ where h_{ii} is the ith dimension of $\mathbf{h}$. The solution of minimizing $\mathbb{Q}$ is

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \frac{1}{2}\left(\| X\mathbf{w} - \mathbf{y}_l \|^2 + \mu \mathbf{w}^T H \mathbf{w} \right) . \tag{3}$$

By differentiating the right-hand side of Eq. (3) with respect to $\mathbf{w}$, we have

$$\mathbf{w}^* = \left(X^T X + \mu H \right)^{-1} X^T \mathbf{y}_l . \tag{4}$$

Now we can directly get the discriminant function $f(x; \mathbf{w})$ through matrix calculation. For more accuracy, one could repeat Eq. (4) until convergence, *i.e.*, thresholding $f(x; \mathbf{w})$ to form new binary $\mathbf{y}_l$ for the next iteration to compute Eq. (4).

2.2. Feature warping

Since our goal is to remove discrete noise and to locate high-density clusters, the choice of feature descriptor $\Psi(\cdot)$ is critical. Here we introduce two simple but effective feature warping methods to distinguish noise and clusters respectively. At first, edge matrix W is computed on original features. Then warping is generated from the normalized graph Laplacian $L = I - S$.

2.2.1. Feature warping for noise removal. As discussed in Section 2.1, an ideal $\Psi(\cdot)$ should have relative smaller values nearly in all dimensions. Thus, noise removal could be conducted by thresholding data with small $f(x; \mathbf{w})$. We follow the viewpoint in [6] to regard noise as discrete points which do not reside in any clusters or manifold shapes, and compute the warping

$$\Psi_1 : \mathcal{X} \to \mathbb{R}^n,\ x_i \mapsto L^{-1}(i, \cdot)^T ,\ i = 1, \cdots n , \tag{5}$$

where $L^{-1}(i, \cdot)$ denotes the ith row vector of the inverse graph Laplacian. Ψ_1 maps x_i to a space of dimension equal to the number of data, and each dimension $\Psi_1(x_i)_j$ indicates the possibility for node i and j resided in a same cluster. The dimension weight diagonal matrix H_1 could be defined as $h_{ii} = 1/\sum_{j=1}^{n} \Psi_1(x_i)_j$. By computing Eq. (4) and Eq. (1), discrete noise would have relative smaller values in $f(x; \mathbf{w})$.

2.2.2. Feature warping for dominant cluster selection. A desirable warping $\Psi(\cdot)$ in this task should

make data in a same cluster more similar. Therefore data in high-density clusters would keep relative larger $f(x;\mathbf{w})$ when minimizing $\mathbb{Q}$. As described in [9], the few top eigenvectors of graph Laplacian show apparent block structure to tell clusters. Let $\{(\mathbf{v_i}, \lambda_\mathbf{i})_{i=1}^n\}$ be eigenvector and eigenvalue pairs of L where $\lambda_1 \leq \cdots \leq \lambda_n$. Top k eigenvectors of L is to form $U_k = [\mathbf{v_1}, \mathbf{v_2}, \cdots, \mathbf{v_k}]$. Our proposed feature warping is

$$\Psi_2 : \mathcal{X} \to \mathbb{R}^k,\ x_j \mapsto U_k(j,\cdot)^T,\ j = 1, \cdots l\ . \quad (6)$$

We only use l labeled queries to estimate parameter $\mathbf{w}$, according to a fact that positive data is usually on higher proportion in query set than in the entire dataset. The dimension weight diagonal matrix H_2 could be defined as $h_{ii} = \lambda_i$, as smaller eigenvalues and eigenvectors are more representative to tell clusters.

2.3. Spectral ranking

In summary, we use Ψ_1 and H_1 to learn $f_1 = \langle \mathbf{w}_1, \Psi_1 \rangle$ and remove noise by thresholding smaller $f_1(x;\mathbf{w})$. Then $f_2 = \langle \mathbf{w}_2, \Psi_2 \rangle$ is learnt from Ψ_2 and H_2 to select data in high-density clusters by thresholding larger $f_2(x;\mathbf{w})$. The selected data is regarded as pure positive queries for the general *SSL* framework

$$\mathbf{g} = (I - \alpha S)^{-1}\mathbf{y}, \quad (7)$$

where $\mathbf{y}$ is the query vector, $\mathbf{g}$ is the classification score, and α is a balance between data smoothness and empirical risk. More details about Eq. (7) could be found in [13]. It is a widely used learning model for image retrieval, graph cut, etc. We experiment on such various applications to show performance of our algorithm.

3. Experiments

We introduce a toy data to show the shortcomings of recent approaches [10, 7], and present our solution in detail. It is notable that [10, 7] have low performance on this specific data due to their great weakness. Then we compare the practicality of all methods on two real applications, image re-ranking and co-segmentation.

3.1. Toy data

The two-circle is combined with discrete noise to form a 300-point dataset, with the ratio of 112:96:92 for inner, outer circle and noise. More points and smaller radius make inner circle the higher-density cluster. We sample 30 points as initial noisy queries (blue markers in Figure 1a) with the ratio of 12:7:11 for inner, outer circle and noise. Our goal is to discriminate points in the inner circle. We compare our algorithm with three existing *SSL* methods. *LGC* [13] directly uses the noisy queries for transductive inference. LabelDiag [10] and SpecFilter [7] are two noise resistant methods. The experimental results are shown in Figure 1 and Table 1.

Table 1. Query set precision and average precision for all methods on toy data experiments.

Toy Data Exp.	Query Prec. (%)	AP (%)
LGC [13]	40.00	41.34
LabelDiag [10]	40.00	45.74
SpecFilter [7]	0.00	28.01
Ours	**100.00**	**100.00**

In Figure 1, the green markers in (b)(c)(d) present the query refining results by each method. It is notable that [10] and [7] fail to extract a pure query set. LabelDiag [10] is ineffective when the precision of initial queries is low (40% in this experiment). SpecFilter [7] is quite sensitive to local compact regions in query set, thus the 5-point small cluster is selected and leads to the worst result. Meanwhile, our method first remove the global scattered noise ("cross" markers in (d)) and then locate points in the most dominant cluster to learn a discriminant function which achieve the highest precision.

3.2. Web image search re-ranking

Our algorithm is well suitable for web image re-ranking when search engine retrieves images based on keywords. The retrieved images are often full of noise while the in-class images are likely to form the highest-density cluster. We use the initial top-50 images as pseudo queries to re-rank, and test the performance on a public database INRIA [4] with 353 categories and 71,478 images. We adopt *LLC* [11] to extract visual features for each image. In our experiments, all methods share the **same** features and the **same** initial queries. The accuracy of query purification and image re-ranking is shown in Table 3. Our algorithm outperforms the others in such a general database. Our global noise removal performance is in Table 2. Re-ranking results on several queries are shown in Figure 2.

3.3. Image co-segmentation

Given several images, co-segmentation is to segment concurrent foreground objects from various background. The features of foreground objects are expected to form high-density clusters which is appropriate for our algorithm to select and segment.

Our implementation is as follows. (1) Each image is over-segmented into several parts by MeanShift [2]. (2) Each part is represented by a visual descriptor. Here we linearly combine texture features [8] and co-saliency map [5] for every pixel then compute bag-of-words histogram for each part. (3) Our algorithm is used to select

Table 2. Image in-class vs. non-class statistics of INRIA database, before and after our noise removal process.

INRIA Exp.	in-class	non-class	prec.
before denoise	31347	40131	43.86%
after denoise	22013	21795	50.25%

Table 3. Query set precision and re-ranking mean average precision for all methods on INRIA database, with top-50 images of each category as noisy queries.

INRIA Exp.	Query Prec. (%)	MAP (%)
Search Engine [4]	56.94	56.99
LGC [13]	56.94	69.46
LabelDiag [10]	56.82	70.12
SpecFilter [7]	60.83	73.58
Ours	**70.72**	**75.10**

some image parts as input accurate foreground queries for graph cut similar to Eq. (7). Examples of segment results on MSRC database [3] are in Figure 3.

4. Conclusion

In this paper, we proposed a simple and effective model to eliminate discrete noise and select data in the most dominant cluster. Experimental results in image re-ranking and co-segmentation demonstrate that our algorithm is a powerful tool to handle various problems in computer vision if the data distribution is applied to our model. In our further research, we will modify our model for other appropriate application scenario.

5. Acknowledgements

This work was supported by Introduced Innovative R&D Team of Guangdong Province (Robot and Intelligent Information Technology); Natural Science Foundation of China (60903117); and Science, Industry, Trade, and Information Technology Commission of Shenzhen Municipality, China (JC201005270357A, ZYC201006130314A).

References

[1] Y. Boykov and M. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *ICCV*, 2001.

[2] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. on PAMI*, 24(5):603–619, 2002.

[3] A. Criminisi. Microsoft research cambridge object recognition image dataset. version 1.0, 2004.

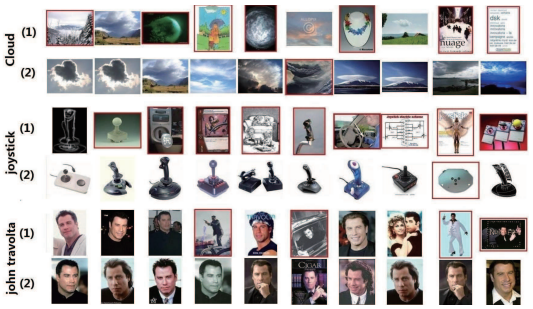

Figure 2. Three re-ranking results in INRIA database. Top 10 images using (1) the search engine [4] (2) our re-ranking approach. Non-class images are in red box.

Figure 3. Four image pairs in MSRC database to show our co-segmentation performance.

[4] J. Krapac, M. Allan, J. Verbeek, and F. Juried. Improving web image search results using query-relative classifiers. In *CVPR*, 2010.

[5] H. Li and K. Ngan. A co-saliency model of image pairs. *IEEE Transactions on Image Processing*, (99):1, 2010.

[6] Z. Li, J. Liu, S. Chen, and X. Tang. Noise robust spectral clustering. In *ICCV*, 2007.

[7] W. Liu, Y. Jiang, J. Luo, and S. Chang. Noise resistant graph ranking for improved web image search. In *CVPR*, 2011.

[8] C. Schmid. Constructing models for content-based image retrieval. In *CVPR*, 2001.

[9] T. Shi, M. Belkin, and B. Yu. Data spectroscopy: Eigenspaces of convolution operators and clustering. *The Annals of Statistics*, 37(6B):3960–3984, 2009.

[10] J. Wang, Y. Jiang, and S. Chang. Label diagnosis through self tuning for web image search. In *CVPR*, 2009.

[11] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *CVPR*, 2010.

[12] Y. Yang, D. Xu, F. Nie, J. Luo, and Y. Zhuang. Ranking with local regression and global alignment for cross media retrieval. In *ACM Multimedia*, 2009.

[13] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In *NIPS*, 2004.