Learning with Inadequate and Incorrect Supervision

Chen Gong*, Hengmin Zhang*, Jian Yang* and Dacheng Tao†
*School of Computer Science and Engineering, Nanjing University of Science and Technology, China
†UBTECH Sydney AI Centre, SIT, FEIT, University of Sydney, Australia
Corresponding author: Chen Gong (E-mail: chen.gong@njust.edu.cn)

***Abstract*—Practically, we are often in the dilemma that the labeled data at hand are inadequate to train a reliable classifier, and more seriously, some of these labeled data may be mistakenly labeled due to the various human factors. Therefore, this paper proposes a novel semi-supervised learning paradigm that can handle both label insufficiency and label inaccuracy. To address label insufficiency, we use a graph to bridge the data points so that the label information can be propagated from the scarce labeled examples to unlabeled examples along the graph edges. To address label inaccuracy, Graph Trend Filtering (GTF) and Smooth Eigenbase Pursuit (SEP) are adopted to filter out the initial noisy labels. GTF penalizes the ℓ_0 norm of label difference between connected examples in the graph and exhibits better local adaptivity than the traditional ℓ_2 norm-based Laplacian smoother. SEP reconstructs the correct labels by emphasizing the leading eigenvectors of Laplacian matrix associated with small eigenvalues, as these eigenvectors reflect real label smoothness and carry rich class separation cues. We term our algorithm as "Semi-supervised learning under Inadequate and Incorrect Supervision" (SIIS). Thorough experimental results on image classification, text categorization, and speech recognition demonstrate that our SIIS is effective in label error correction, leading to superior performance to the state-of-the-art methods in the presence of label noise and label scarcity.**

***Index Terms*—semi-supervised learning; label noise; graph trend filtering; smooth eigenbase pursuit**

I. Introduction

Practically, it is quite often that the available labeled data are insufficient for training a reliable supervised classifier such as Support Vector Machines (SVM) and Convolutional Neural Networks (CNN). For example, manually annotating web-scale images/texts is intractable because of the unacceptable human labor cost. Acquiring sufficient labeled examples for protein structure categorization is also infeasible as it often takes months of laboratory work for experts to identify a single protein's 3D structure. To make the matter worse, a portion of such limited labeled data are very likely to be mislabeled, which means that the sparse supervision information we have may not be reliable. For example, in crowdsourced image annotation, some of the image labels can be incorrect due to the knowledge or cultural limitation of the annotators. The labeling of protein structure is also error-prone as this process is highly depended on the experience and expertise of labelers working in the biological area.

To solve the abovementioned practical problems, this paper studies how to leverage the scarce labeled examples with untrustable labels to build a reliable classifier so that the massive unlabeled examples can be accurately classified. Therefore, two issues are jointly taken into consideration in this paper: one is the insufficiency of labeled examples, and the other is the noise in label space.

In fact, Semi-Supervised Learning (SSL) [1] has been widely used to deal with the first issue. SSL aims to predict the labels of a large amount of unlabeled examples given only a few labeled examples, and the algorithms of SSL can be roughly divided into three categories, i.e. *collaboration-based*, *large-margin-based*, and *graph-based*. Collaboration-based methods usually contain multiple learners and they are trained collaboratively to improve the integrated performance on the unlabeled data. Co-training [2] and Tri-training [3] are representative methodologies belonging to this category. Large-margin-based methods assume that there exists an optimal decision boundary in the low density region between data clusters, so that the margin between the decision boundary and the nearest data points on each side can be maximized. The algorithms based on the large-margin assumption are usually the variants of traditional SVM, such as Semi-Supervised SVM (S3VM) [4], Mean S3VM [5], and Safe S3VM (S4VM) [6]. Graph-based methods are usually established on the manifold assumption, namely the entire dataset contains a potential manifold, and the labels of examples should vary smoothly along this manifold. The representative algorithms include [7], [8], [9], [10]. All the above SSL methods are not applicable to the label noise situations, which means their performance will significantly decrease in the presence of mislabeled examples.

For the issue of label noise, several works have been done to prevent the performance degradation caused by such incorrect supervision. By discovering that most of the existing loss functions can be decomposed as a label-independent term plus a label-dependent term, Gao et al. [11] and Patrini et al. [12] estimate the unbiased geometric mean of the entire dataset to suppress the negative influence of noisy labels. More generally, a variety of methods have been proposed to adapt the existing loss functions to the corrupted labels via weighting [13], [14], [15], calibration [16], or upper bounding [17]. However, they are designed for supervised classifier and thus are not suitable for the SSL problem considered in this paper.

Therefore, in this paper we aim to design a semi-supervised algorithm that is robust to label noise, so that the unlabeled examples can be accurately classified although the labels at hand might be scarce and inaccurate. Consequently, our method is termed "**S**emi-supervised learning under **I**nadequate and **I**ncorrect **S**upervision" (SIIS). Mathematically, suppose we have l labeled examples $\mathcal{L} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ and u unlabeled examples $\mathcal{U} = \{\mathbf{x}_i\}_{i=l+1}^{n}$ with $n = l + u$. The labels y_i for

$1 \leq i \leq l$ take values from $\{-1, 1\}$ and some of them may be incorrect, then our goal is to classify the examples in $\mathcal{U}$ based on the inaccurate $\mathcal{L}$. Specifically, we build a graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ where $\mathcal{V}$ is the vertex set consisted of all n examples, and $\mathcal{E}$ is the edge set encoding the similarity between these examples. Based on $\mathcal{G}$, we adopt two measures to deal with the possible label noise, namely Graph Trend Filtering (GTF) and Smooth Eigenbase Pursuit (SEP).

GTF [18] is a statistical method to conduct the nonparametric regression on a graph. Its main idea is to penalize the ℓ_0 norm of label difference between graph vertices rather than using the usual ℓ_2 norm-based graph Laplacian smoother [8], [19], [20]. Consequently, the label difference between the connected vertices can be exactly zero by employing GTF. In contrast, the ℓ_2 norm-based Laplacian smoother only decides the vertex difference to be small or large and can hardly set any label difference to exact zero. Therefore, GTF has a stronger power on correcting the noisy labels and achieves better local adaptivity than the traditional Laplacian smoother. SEP stems from the spectral graph theory [1], which claims that the eigenvectors of graph Laplacian matrix corresponding to the smallest eigenvalues reflect the real underlying smoothness of labels and usually contain clear indication of class separations. Thanks to the collaboration of GTF and SEP, our SIIS model performs robustly to the label noise, which will be demonstrated by the experiments.

II. Our Model

In our method, we construct a K-nearest neighborhood (KNN) graph $\mathcal{G}$ over $\mathcal{L} \cup \mathcal{U}$, which is further quantified by the adjacency matrix $\mathbf{W}$. The (i, j)-th element of $\mathbf{W}$, i.e. W_{ij}, encodes the similarity between the examples $\mathbf{x}_i$ and $\mathbf{x}_j$, which is computed by $W_{ij} = \exp\left(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / (2\xi^2)\right)$ with ξ being the Gaussian kernel width if $\mathbf{x}_i$ and $\mathbf{x}_j$ are linked by an edge in $\mathcal{G}$, and $W_{ij} = 0$ otherwise. Based upon $\mathbf{W}$, we introduce the diagonal degree matrix $\mathbf{D}_{ii} = \sum_{j=1}^{n} W_{ij}$ and graph Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{W}$. Besides, we use an l-dimensional vector $\mathbf{y} = (y_1, y_2, \cdots, y_l)^\top$ to record the given labels of l initial labeled examples, and employ an n-dimensional vector $\mathbf{f} = (f_1, \cdots, f_l, f_{l+1}, \cdots, f_n)^\top$ with $\{f_i\}_{i=1}^{n} \in \mathbb{R}$ to represent the soft labels of all n examples. Furthermore, we define a $|\mathcal{E}| \times n$ ($|\mathcal{E}|$ is the size of edge set $\mathcal{E}$) matrix $\mathbf{P}$, in which the k-th ($1 \leq k \leq |\mathcal{E}|$) row corresponds to the edge k that connects $\mathbf{x}_i$ and $\mathbf{x}_j$. Specifically, the k-th row is defined by

$$\mathbf{P}_{k,:} = \begin{pmatrix} 0 & \cdots & \underset{\uparrow}{W_{ij}} & \cdots & \underset{\uparrow}{-W_{ij}} & \cdots & 0 \end{pmatrix}. \begin{matrix} \\ \\ i \\ \\ j \\ \\ \end{matrix} \tag{1}$$

Therefore, the *Graph Trend Filtering Term* (*GTF term*) is expressed as $\|\mathbf{Pf}\|_0 = \sum_{(i,j)\in\mathcal{E}} W_{ij} \mathbb{1}[\![f_i \neq f_j]\!]$ with "$\mathbb{1}[\![\cdot]\!]$" being the indicator function. Our model is then formulated as

$$\min_{\mathbf{f}=(f_1,\cdots,f_n)^\top} \|\mathbf{Pf}\|_0 + \alpha \|\mathbf{Jf} - \mathbf{y}\|_0, \tag{2}$$

where $\alpha > 0$ is the trade-off parameter, and "$\|\cdot\|_0$" represents the ℓ_0 norm that counts the non-zero elements in the corresponding vector; $\mathbf{J}$ is an $l \times n$ matrix with the (i, i)-th ($i = 1, 2, \cdots, l$) elements being 1, and the other elements being 0. The first *GTF term* in (2) enforces the strongly connected examples to obtain identical labels. The second term is *fidelity term* which requires that the optimized $\mathbf{f}$ on the initial labeled examples should approach to the given labels in $\mathbf{y}$. However, the inconsistency between f_i and y_i is allowed due to the adopted ℓ_0 norm, as not all the elements in $\mathbf{y}$ are correct and trustable.

Furthermore, since the Laplacian matrix $\mathbf{L}$ is semi-positive definite, it can be decomposed as $\mathbf{L} = \bar{\mathbf{U}}\bar{\mathbf{\Sigma}}\bar{\mathbf{U}}^\top$ where $\bar{\mathbf{\Sigma}} = diag(\lambda_1, \lambda_2, \cdots, \lambda_n)$ is a diagonal matrix with $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ being the totally n eigenvalues, and $\bar{\mathbf{U}} = (\mathbf{U}_1, \mathbf{U}_2, \cdots, \mathbf{U}_n)$ contains the n associated eigenvectors. Since $\mathbf{U}_1, \mathbf{U}_2, \cdots, \mathbf{U}_n$ are orthogonal, all possible $\mathbf{f}$ on the graph $\mathcal{G}$ can be represented by $\mathbf{f} = \sum_{i=1}^{n} a_i \mathbf{U}_i$ where $\{a_i\}_{i=1}^{n}$ are representation coefficients. According to [1], [21], the first m (typically $m \ll n$) eigenvectors usually depict the label smoothness and convey the real class separation, so they can be employed to reconstruct the optimal $\mathbf{f}$ and meanwhile filter out the incorrect initial labels. Therefore, we may write $\mathbf{f} = \mathbf{Ua}$ in (2) where $\mathbf{U} = (\mathbf{U}_1, \cdots, \mathbf{U}_m)$ is the sub-matrix of $\bar{\mathbf{U}}$ containing the first m columns of $\bar{\mathbf{U}}$, and $\mathbf{a}$ is the corresponding coefficient vector, so we have

$$\min_{\mathbf{a}=(a_1,\cdots,a_m)^\top} \|\mathbf{PUa}\|_0 + \alpha \|\mathbf{JUa} - \mathbf{y}\|_0 + \beta \mathbf{a}^\top \mathbf{\Sigma a}, \tag{3}$$

where $\mathbf{\Sigma}$ is a diagonal matrix with $\mathbf{\Sigma} = diag(\lambda_1, \cdots, \lambda_m)$, $\alpha, \beta > 0$ are two trade-off parameters, and $\mathbf{a}$ is the coefficient vector to be optimized. In (3), the first two terms are directly adapted from (2). The third term allows the coefficient a_i to be large if the corresponding eigenvalue λ_i is small, which means that the eigenbasis $\mathbf{U}_i$ with smaller eigenvalues λ_i are preferred in the reconstruction of the optimal $\mathbf{f}$, as they are usually smooth and contain rich class information. In contrast, the value of a_i should be suppressed to a small value if the associated eigenvalue λ_i is large.

Considering that (3) involves ℓ_0 norm that is usually difficult to optimize, we may replace the ℓ_0 norm with the surrogate ℓ_1 norm which computes the sum of absolute values of the vector's elements. To handle multi-class cases, the SIIS model for binary classification can be further extended to

$$\min_{\mathbf{A}\in\mathbb{R}^{m\times c}} \|\mathbf{PUA}\|_{2,1} + \alpha \|\mathbf{JUA} - \mathbf{Y}\|_{2,1} + \beta \mathrm{tr}\left(\mathbf{A}^\top \mathbf{\Sigma A}\right), \tag{4}$$

in which $\|\mathbf{H}\|_{2,1}$ calculates the $\ell_{2,1}$ norm of the matrix $\mathbf{H}$ by $\|\mathbf{H}\|_{2,1} = \sum_i \sqrt{\sum_j H_{ij}^2}$ [22]. $\mathbf{Y} \in \{0, 1\}^{l\times c}$ (c is the total number of classes) is the label matrix of the initial labeled examples, of which the i-th row $\mathbf{Y}_{i,:}$ indicates the label of $\mathbf{x}_i \in \mathcal{L}$. To be specific, $\mathbf{Y}_{ij} = 1$ if $\mathbf{x}_i$ belongs to the j-th class, and 0 otherwise. Therefore, the "clean" soft label matrix of all the examples in $\mathcal{L} \cup \mathcal{U}$ can be recovered by $\mathbf{F} = \mathbf{UA}$ in which the (i, j)-th element F_{ij} represents the posterior probability of $\mathbf{x}_i$ belonging to the j-th class. Consequently, the example $\mathbf{x}_i \in \mathcal{L} \cup \mathcal{U}$ is classified into the j-th class if $j = \arg\max_{j'\in\{1,\cdots,c\}} F_{ij'}$.

III. Optimization

Without loss of generality, this section introduces the optimization for model (4). By letting $\mathbf{Q} = \mathbf{PUA}$ and $\mathbf{B} =$

$\mathbf{JUA}-\mathbf{Y}$, (4) can be transformed to

$$\begin{aligned}\min_{\mathbf{A},\mathbf{B},\mathbf{Q}}\ & \|\mathbf{Q}\|_{2,1}+\alpha\|\mathbf{B}\|_{2,1}+\beta\mathrm{tr}\left(\mathbf{A}^\top\mathbf{\Sigma}\mathbf{A}\right)\\ s.t.\ & \mathbf{Q}=\mathbf{PUA},\ \mathbf{B}=\mathbf{JUA}-\mathbf{Y}.\end{aligned}\tag{5}$$

This constrained optimization problem can be easily solved by using the Alternating Direction Method of Multipliers (ADMM), which alternatively optimizes one variable at one time with the other variables remaining fixed. The augmented Lagrangian function is

$$\begin{aligned}L(\mathbf{A},\mathbf{B},\mathbf{Q},\mathbf{\Lambda}_1,\mathbf{\Lambda}_2,\mu)=&\|\mathbf{Q}\|_{2,1}+\alpha\|\mathbf{B}\|_{2,1}+\beta\mathrm{tr}\left(\mathbf{A}^\top\mathbf{\Sigma}\mathbf{A}\right)\\&+\mathrm{tr}\left(\mathbf{\Lambda}_1^\top(\mathbf{Q}-\mathbf{PUA})\right)+\mathrm{tr}\left(\mathbf{\Lambda}_2^\top(\mathbf{B}-\mathbf{JUA}+\mathbf{Y})\right)\\&+\frac{\mu}{2}\left(\|\mathbf{Q}-\mathbf{PUA}\|_{\mathrm{F}}^2+\|\mathbf{B}-\mathbf{JUA}+\mathbf{Y}\|_{\mathrm{F}}^2\right),\end{aligned}\tag{6}$$

where $\mathbf{\Lambda}_1$ and $\mathbf{\Lambda}_2$ are Lagrangian multipliers, $\mu>0$ is the penalty coefficient, and "$\|\cdot\|_{\mathrm{F}}$" denotes the Frobenius norm of the corresponding matrix. Based on (6), the variables $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{Q}$ can be sequentially updated via an iterative way.

Update Q: The subproblem related to $\mathbf{Q}$ is

$$\begin{aligned}&\min_{\mathbf{Q}}\|\mathbf{Q}\|_{2,1}+\mathrm{tr}\left(\mathbf{\Lambda}_1^\top(\mathbf{Q}-\mathbf{PUA})\right)+\frac{\mu}{2}\|\mathbf{Q}-\mathbf{PUA}\|_{\mathrm{F}}^2\\&\Leftrightarrow\frac{1}{\mu}\|\mathbf{Q}\|_{2,1}+\frac{1}{2}\|\mathbf{Q}-\mathbf{N}\|_{\mathrm{F}}^2,\end{aligned}\tag{7}$$

where $\mathbf{N}=\mathbf{PUA}-\frac{1}{\mu}\mathbf{\Lambda}_1$.

According to [23], the solution of (7) can be expressed as

$$\mathbf{Q}_{i,:}=\begin{cases}\dfrac{\|\mathbf{N}_{i,:}\|_2+1/\mu}{\|\mathbf{N}_{i,:}\|_2}\mathbf{N}_{i,:}, & 1/\mu<\|\mathbf{N}_{i,:}\|_2\\ 0, & \text{otherwise}\end{cases},\tag{8}$$

where $\mathbf{N}_{i,:}$ represents the i-th row of matrix $\mathbf{N}$.

Update B: By dropping the unrelated terms to $\mathbf{B}$ in (6), the subproblem regarding $\mathbf{B}$ is

$$\min_{\mathbf{B}}\alpha\|\mathbf{B}\|_{2,1}+\mathrm{tr}\left(\mathbf{\Lambda}_2^\top(\mathbf{B}-\mathbf{JUA}+\mathbf{Y})\right)+\frac{\mu}{2}\|\mathbf{B}-\mathbf{JUA}+\mathbf{Y}\|_{\mathrm{F}}^2,\tag{9}$$

which is equivalent to

$$\min_{\mathbf{B}}\frac{\alpha}{\mu}\|\mathbf{B}\|_{2,1}+\frac{1}{2}\|\mathbf{B}-\mathbf{M}\|_{\mathrm{F}}^2,\tag{10}$$

where $\mathbf{M}=\mathbf{JUA}-\mathbf{Y}-\frac{1}{\mu}\mathbf{\Lambda}_2$. Similar to (8), the optimizer of (10) is

$$\mathbf{B}_{i,:}=\begin{cases}\dfrac{\|\mathbf{M}_{i,:}\|_2+\alpha/\mu}{\|\mathbf{M}_{i,:}\|_2}\mathbf{M}_{i,:}, & \alpha/\mu<\|\mathbf{M}_{i,:}\|_2\\ 0, & \text{otherwise}\end{cases}.\tag{11}$$

Update A: The subproblem regarding $\mathbf{A}$ is

$$\begin{aligned}\min_{\mathbf{A}}\ &\beta\mathrm{tr}\left(\mathbf{A}^\top\mathbf{\Sigma}\mathbf{A}\right)-\mathrm{tr}\left(\mathbf{\Lambda}_1^\top\mathbf{PUA}\right)-\mathrm{tr}\left(\mathbf{\Lambda}_2^\top\mathbf{JUA}\right)\\&+\frac{\mu}{2}\left(\|\mathbf{Q}-\mathbf{PUA}\|_{\mathrm{F}}^2+\|\mathbf{B}-\mathbf{JUA}+\mathbf{Y}\|_{\mathrm{F}}^2\right).\end{aligned}\tag{12}$$

By computing the derivative of (12) w.r.t. $\mathbf{A}$, and then setting the result to zero, we have

$$\begin{aligned}\mathbf{A}=&\left(2\beta\mathbf{\Sigma}+\mu\mathbf{U}^\top\mathbf{P}^\top\mathbf{PU}+\mu\mathbf{U}^\top\mathbf{J}^\top\mathbf{JU}\right)^{-1}\\&\left[\mathbf{U}^\top\mathbf{P}^\top\mathbf{\Lambda}_1+\mathbf{U}^\top\mathbf{J}^\top\mathbf{\Lambda}_2+\mu\mathbf{U}^\top\mathbf{P}^\top\mathbf{Q}+\mu\mathbf{U}^\top\mathbf{J}^\top(\mathbf{B}+\mathbf{Y})\right].\end{aligned}\tag{13}$$

The entire proposed SIIS algorithm is summarized in Algorithm 1, in which the ADMM process is guaranteed to converge according to the result in [24].

Algorithm 1 Summary of the proposed algorithm.

Input: α, β, m, K, $\mathbf{Y}$.
Construct KNN graph and compute the adjacency matrix $\mathbf{W}$;
Compute the m smallest eigenvalues and eigenvectors of Laplacian matrix $\mathbf{L}$, and store them in $\mathbf{\Sigma}$ and $\mathbf{U}$, respectively;
// Begin classification
Set $\mathbf{\Lambda}_1,\mathbf{\Lambda}_2$ to all-one matrices; Initialize $\mathbf{A}=\mathbf{O}$; Set $\mu=1$, $\mu_{max}=10^{10}$, $\rho=1.2$, $\epsilon=10^{-4}$, $MaxIter=100$;
set $iter=0$;
repeat
  Update $\mathbf{Q}$ via (8);
  Update $\mathbf{B}$ via (11);
  Update $\mathbf{A}$ via (13);
  // Update Lagrangian multipliers
  $\mathbf{\Lambda}_1:=\mathbf{\Lambda}_1+\mu(\mathbf{Q}-\mathbf{PUA})$, $\mathbf{\Lambda}_2:=\mathbf{\Lambda}_2+\mu(\mathbf{B}-\mathbf{JUA}+\mathbf{Y})$;
  // Update penalty coefficients
  $\mu:=\min(\rho\mu,\mu_{max})$;
  $iter:=iter+1$;
until $\frac{\|\mathbf{A}^{(iter)}-\mathbf{A}^{(iter-1)}\|_\infty}{\|\mathbf{A}^{(iter-1)}\|_\infty}\le\epsilon$ or $iter=MaxIter$
Recover $\mathbf{F}=\mathbf{UA}$;
Classify $\mathbf{x}_i\in\mathcal{L}\cup\mathcal{U}$ to the j-th class via $j=\arg\max_{j'\in\{1,\cdots,c\}}F_{ij'}$;
Output: Class labels $\{y_i\}_{i=1}^n$.

IV. Experimental Results

In this section, we first validate the motivation of the proposed algorithm (Section IV-A), and then compare our SIIS with several representative approaches on various practical datasets (Sections IV-B∼IV-D). Finally, we study the parametric sensitivity of SIIS (Section IV-E).

A. Algorithm Validation

There are three critical components in our model (3) for tackling the possible label noise: 1) ℓ_0 norm is adopted to form the fidelity term, allowing the obtained solution to be slightly inconsistent with the initial given labels in $\mathbf{y}$; 2) GTF term is employed to adaptively correct the possible label error in a local area, which is better than the global smoothness term $\mathrm{tr}(\mathbf{f}^\top\mathbf{Lf})$ that has been widely used by many existing methodologies; and 3) The SEP strategy is leveraged to recover the precise labels by emphasizing the $\mathbf{L}$'s leading eigenvectors with clear class indication. Therefore, here we use a two-dimensional toy dataset, i.e. *NoisyDoubleMoon*, to visually show the strength of each of above three components.

NoisyDoubleMoon consists of 640 examples, which are equally divided into two moons. This dataset is contaminated by the Gaussian noise with standard deviation 0.15, and each class has three initial labeled examples (see Fig. 1(a)). However, each class contains one erroneously labeled example (marked by the purple circle), which poses a great difficulty for an algorithm to achieve perfect classification.

Fig. 1(b) presents the result of Gaussian Field and Harmonic Functions (GFHF) [7], of which the model is $\min_{\mathbf{f}:\mathbf{f}_\mathcal{L}=\mathbf{y}}\mathrm{tr}(\mathbf{f}^\top\mathbf{Lf})$. As GFHF requires the finally obtained $\mathbf{f}$ to be strictly identical to $\mathbf{y}$ on labeled set $\mathcal{L}$ and completely ignores the label noise, we see that it is greatly misled by the two incorrect labels and only obtains 78.01% accuracy. In (c), we replace the equality constraint $\mathbf{f}_\mathcal{L}=\mathbf{y}$

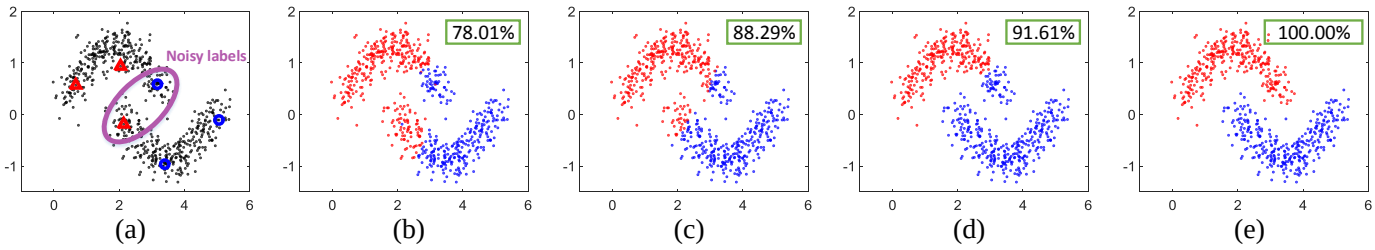

Fig. 1. Algorithm validation on *NoisyDoubleMoon* dataset. (a) shows the initial state with labeled positive examples (red triangles) and labeled negative examples (blue circles). Note that two examples are mislabeled and they form the label noise. (b)∼(e) present the classification results of different settings.

in GFHF with a robust ℓ_0 fidelity term, and the model is $\min_{\mathbf{f}} \ \mathrm{tr}(\mathbf{f}^\top \mathbf{L}\mathbf{f}) + \alpha \|\mathbf{J}\mathbf{f} - \mathbf{y}\|_0$. We observe that this ℓ_0 fidelity term greatly weakens the negative effect of mislabeled examples, and the performance can be improved to 88.29%, therefore the effectiveness of component 1) is verified. In (d), we further replace the traditional Laplacian smoother $\mathrm{tr}(\mathbf{f}^\top \mathbf{L}\mathbf{f})$ in (c) with GTF term to form the model $\min_{\mathbf{f}} \ \|\mathbf{P}\mathbf{f}\|_0 + \alpha \|\mathbf{J}\mathbf{f} - \mathbf{y}\|_0$, and investigate the effect brought by GTF. It can be clearly observed that the false-positive labeled data point in the below moon has been corrected, and thus all negative examples have been successfully identified. Consequently, the classification accuracy has been improved to 91.61%, which means that GTF is helpful for eliminating the label noise. Finally, we illustrate the classification result yielded by the proposed model in (e). By preserving the top-2 eigenvectors of $\mathbf{L}$ to reconstruct $\mathbf{f}$, all examples are accurately classified, therefore component 3) contributes to enhance the robustness of our SIIS method. In short, every step included by SIIS is helpful for suppressing the adverse effect of noisy labels.

B. Image Data

We firstly use the *COIL* dataset [25] to test the ability of our SIIS on processing image data. *COIL* is a popular public dataset for object recognition which contains 1440 object images belonging to 20 classes , and each object has 72 images shot from different angles. The resolution of each image is 32×32, with 256 grey levels per pixel. Thus, every image is represented by a 1024-dimensional element-wise vector. We randomly select 10 examples from each class to establish the labeled set, and then the remaining image examples form the unlabeled set. To incorporate different levels of label noise to the dataset, we randomly pick up 0%, 20%, 40% and 60% examples from the totally $10 \times 20 = 200$ labeled examples, and switch the correct label of each of them to a random wrong label. Such label contamination is conducted 10 times, so every compared algorithm should independently run 10 times on the contaminated dataset and the average accuracy over these 10 different runs are particularly investigated.

The compared algorithms include: 1) the traditional supervised classifier SVM; 2) typical SSL method GFHF [7] which utilizes Laplacian smoother and does not consider the label noise, and 3) state-of-the-art label-noise robust SSL methodologies such as Large-Scale Sparse Coding (LSSC) [26], Graph Trend Filtering (GTF) [18], and Self-Paced Manifold Regularization (SPMR) [27].

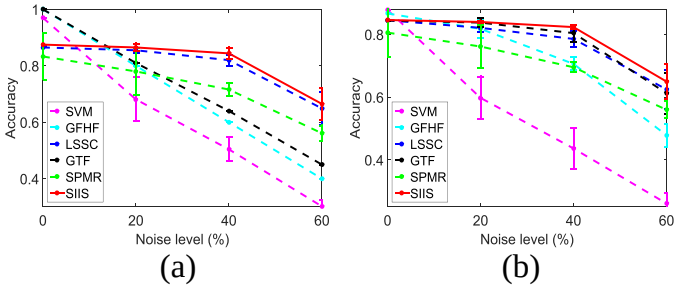

Fig. 2. The comparison of various algorithms on *COIL* dataset. (a) shows their classification accuracies on labeled set, and (b) plots the accuracies on unlabeled set.

For fair comparison, all graph-based algorithms such as GFHF, LSSC, GTF, SPMR and SIIS are implemented on a 10-NN graph with Gaussian kernel width $\xi = 100$. In SIIS, β is set to 10, and α is tuned to 10^5, 10^2, 10^2 and 10^2 when the noise rate is 0%, 20%, 40% and 60%, respectively. The number of preserved eigenvectors is fixed to $m = 30$. Similarly, the trade-off parameter for fidelity term in GTF is also tuned to 100, and the first 30 eigenvectors are employed to reconstruct the labels in LSSC. In SPMR, the parameters are tuned to $\gamma_K = 1$ and $\gamma_I = 0.01$ as suggested by [27]. The weighting parameter in SVM is tuned to 1.

The classification accuracies of all compared methods on labeled set and unlabeled set are presented in Figs. 2(a) and (b), respectively. It can be observed that the performances of all algorithms decrease with the increase of label noise level. However, the proposed SIIS achieves the best results in most cases when compared with other baselines. Another notable fact is that SVM, which is a traditional supervised algorithm, performs worse than any of the SSL methods, therefore SSL adopted in this paper is more effective than supervised models when the supervised information is inadequate. When it comes to SSL approaches, we see that the performance of GFHF decreases dramatically when the noise level ranges from 0% to 60%, which suggests that the ℓ_2 norm-based Laplacian smoother would fail in the presence of contaminated labels. In contrast, some robust SSL methods including LSSC, GTF and our SIIS are very stable although the noise level increases rapidly, especially in the range $[0\%, 40\%]$.

C. Text Data

This section compares the ability of SVM, GFHF, LSSC, GTF, SPMR, and the proposed SIIS on text categorization. Similar to [28], a subset of the *Reuters Corpus Volume 1* (*RCV1*) dataset [29] is adopted for comparison, which contains

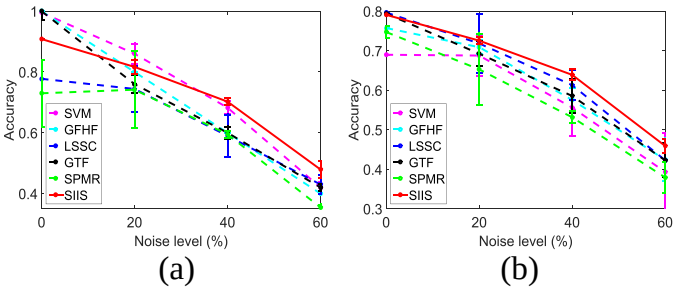

Fig. 3. The comparison of various algorithms on *RCV1* dataset. (a) shows their classification accuracies on labeled set, and (b) displays the accuracies on unlabeled set.

9625 news article examples across four classes (i.e. "C15", "ECAT", "GCAT", and "MCAT"). As there are 29992 distinct words in this dataset, the standard TF-IDF weighting scheme is adopted to generate a 29992-dimensional feature vector for each example. In this dataset, we randomly select 240 examples out of the totally 9625 examples to establish the labeled set, and the remaining 9385 examples are regarded as unlabeled. As a result, the labeled examples only accounts for approximately 2.5% of the entire dataset, leading to the scarcity of the supervised information.

The parameters of established graph are $K = 10$ and $\xi = 10$. The free parameters α and β are tuned via searching the grid $[1, 10, 100, 1000]$, and their values are determined as $\alpha = 1000$ and $\beta = 10$. In Section IV-E, we will explain the reason for choosing such parametric setting. The performances rendered by the compared methods are displayed in Fig. 3. From (a), we see that SVM, GFHF and GTF obtain almost 100% accuracy on the labeled set $\mathcal{L}$ when there is no label noise. However, their accuracies decrease dramatically when we gradually increase the noise level, and they are worse than SIIS under heavy label noise in the range $[40\%, 60\%]$. As a consequence, they are inferior to SIIS in terms of the classification accuracy on unlabeled set $\mathcal{U}$ as reflected by (b). Besides, by comparing the accuracies of SIIS and GFHF when label noise presents, we see that SIIS leads GFHF with a noticeable margin no matter on the labeled set or unlabeled set. This demonstrates that SIIS with GTF term and SEP term is better than the GFHF with ℓ_2 norm-based Laplacian smoother on amending the label errors. Furthermore, by comparing SIIS and GTF, we note that the performance of GTF can be remarkably improved by SIIS, and this again validates that the SEP in our SIIS plays an important role in filtering out the noisy labels.

D. Audio Data

In this experiment, we address a speech recognition task by using the *ISOLET* dataset[1]. In this dataset, 150 subjects are required to pronounce each letter in the alphabet (i.e. "A"$\sim$"Z") twice. Excluding 3 missing examples, we have totally $150 \times 2 \times 26 - 3 = 7797$ examples. Our task is to identify which of the 26 letters every example belongs to. Among the 7797 examples, we extract 40 examples from each class to form the labeled set with size 1040, and the rest 6757 examples are treated as unlabeled.

[1]https://archive.ics.uci.edu/ml/datasets/ISOLET

TABLE I
THE COMPARISON OF VARIOUS METHODS ON *ISOLET* DATASET. THE CLASSIFICATION ACCURACIES ON LABELED EXAMPLES ARE PRESENTED. THE BEST RECORD UNDER EACH LABEL NOISE LEVEL IS MARKED IN BOLD.

Noise level	0%	20%	40%	60%
SVM	0.943±0.000	0.813±0.012	0.687±0.022	0.521±0.015
GFHF	**1.000±0.000**	0.800±0.000	0.600±0.000	0.400±0.000
LSSC	0.899±0.000	0.877±0.003	0.829±0.009	0.718±0.017
GTF	0.958±0.000	0.798±0.007	0.633±0.004	0.553±0.006
SPMR	0.685±0.000	0.638±0.008	0.635±0.004	0.548±0.005
SIIS	0.911±0.000	**0.905±0.008**	**0.836±0.010**	**0.774±0.010**

TABLE II
THE COMPARISON OF VARIOUS METHODS ON *ISOLET* DATASET. THE CLASSIFICATION ACCURACIES ON UNLABELED EXAMPLES ARE PRESENTED. THE BEST RECORD UNDER EACH LABEL NOISE LEVEL IS MARKED IN BOLD.

Noise level	0%	20%	40%	60%
SVM	0.851±0.000	0.805±0.011	0.729±0.021	0.594±0.017
GFHF	**0.865±0.000**	0.816±0.004	0.797±0.010	0.674±0.015
LSSC	0.848±0.000	0.828±0.003	0.785±0.006	0.677±0.018
GTF	0.701±0.000	0.699±0.002	0.598±0.003	0.548±0.005
SPMR	0.689±0.000	0.638±0.008	0.627±0.004	0.539±0.005
SIIS	0.854±0.000	**0.849±0.006**	**0.802±0.013**	**0.749±0.014**

We vary the label noise level from 0% to 60%, and investigate the classification performances of SVM, GFHF, LSSC, GTF, SPMR and SIIS on labeled and unlabeled examples. From Tables I and II, we see that when the noise rate is 0%, our SIIS performs worse than the GFHF and SVM that does not consider the label noise. However, if 20%$\sim$60% labeled examples are erroneously annotated, the advantage of SIIS becomes prominent among all compared algorithms. This is because SIIS is designed under the assumption that the dataset contains the label noise, and the obtained labels on $\mathcal{L}$ is also allowed to be inconsistent with the given labels in $\mathbf{y}$ (see the fidelity term of (2)). Therefore, SIIS generates inferior results to GFHF and SVM when all the examples are correctly labeled, but obtains better performance when the dataset does contain the mislabeled examples. Specifically, SIIS respectively reaches 77.4% accuracy on $\mathcal{L}$ and 74.9% accuracy on $\mathcal{U}$ under 60% noise rate, which leads the second best algorithm LSSC with a gap 5.6% and 7.2% correspondingly. Besides, it can be found that the error rates of GFHF on $\mathcal{L}$ are equivalent to the noise rates such as 20%, 40% and 60%, as this method requires the labels of original labeled examples to remain unchanged during the classification. In contrast, our SIIS generates 90.5%, 83.6% and 77.4% accuracy on $\mathcal{L}$ when 20%, 40% and 60% labels are not correct, which means that 10.5%, 23.6% and 37.4% deteriorated labels have been corrected correspondingly, and this is the reason that our method is able to obtain satisfactory performance on classifying the unlabeled examples in the presence of heavy label noise.

E. Parametric Sensitivity

Note that the objective function (4) in our method contains two trade-off parameters α and β that should be manually

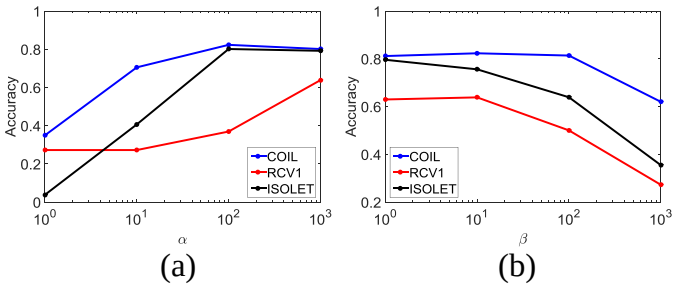

Fig. 4. Parametric sensitivity of the proposed SIIS. (a) and (b) plot the accuracy of SIIS w.r.t. the variation of α and β, respectively.

tuned. Therefore, in this section we discuss whether the choices of them will significantly influence the performance of SIIS. To this end, we examine the classification accuracy of SIIS on unlabeled set by varying one of α and β, and meanwhile fixing the other one to a constant value [30]. The three practical datasets from Sections IV-B to IV-D are adopted including *COIL*, *RCV1*, and *ISOLET*. By changing α and β from 10^0 to 10^3, the results under 40% label noise on the three datasets are shown in Fig. 4. From the experimental results, we learn that the these two parameters are critical for our algorithm to achieve good performance. To be specific, α is suggested to choose a relatively large number, such as $10^2 \sim 10^3$ on *COIL* and *ISOLET* datasets, and 10^3 on *RCV1* dataset. According to Fig. 4(b), we see that a small β is preferred to obtain high accuracy, therefore this parameter is set to 10, 10, and 1 on *COIL*, *RCV1* and *ISOLET*, respectively.

V. Conclusion

To solve the label shortage and label inaccuracy that often occur in many real-world problems, this paper proposed a novel graph-based SSL algorithm dubbed "Semi-supervised learning under Inadequate and Incorrect Supervision" (SIIS). Two measures, namely graph trend filtering and smooth eigenbase pursuit, are formulated into a unified optimization framework to tackle the label errors. We tested our SIIS on image, text and audio datasets under different levels of label noise, and found that SIIS performs robustly to label noise and achieves superior performance to other compared baseline methods. In particular, SIIS is able to obtain very encouraging results when more than half of the limited labeled examples are mislabeled, which further demonstrates the effectiveness and robustness of the proposed algorithm.

Acknowledgment

This research is supported by NSF of China (No: 61602246, 61572315, 91420201, 61472187, 61502235, 61233011 and 61373063), 973 Plan of China (No: 2014CB349303 and 2015CB856004), Program for Changjiang Scholars, NSF of Jiangsu Province (No: BK20171430), the Postgraduate Research & Practice Innovation Program of Jiangsu Province, and Australian Research Council Discovery Project (No: FL-170100117, DP-140102164, LP-150100671).

References

[1] X. Zhu and B. Goldberg, *Introduction to Semi-Supervised Learning*. Morgan & Claypool Publishers, 2009.

[2] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *COLT*. ACM, 1998, pp. 92–100.

[3] Z. Zhou and M. Li, "Tri-training: Exploiting unlabeled data using three classifiers," *TKDE*, vol. 17, no. 11, pp. 1529–1541, 2005.

[4] T. Joachims, "Transductive inference for text classification using support vector machines," in *ICML*, vol. 99, 1999, pp. 200–209.

[5] Y. Li, J. Kwok, and Z. Zhou, "Semi-supervised learning using label mean," in *ICML*. ACM, 2009, pp. 633–640.

[6] Y. Li and Z. Zhou, "Towards making unlabeled data never hurt," *TPAMI*, vol. 37, no. 1, pp. 175–188, 2015.

[7] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using Gaussian fields and harmonic functions," in *ICML*, Washington, DC, USA, 2003.

[8] D. Zhou and O. Bousquet, "Learning with local and global consistency," in *NIPS*, 2003, pp. 321–328.

[9] F. Wang and C. Zhang, "Label propagation through linear neighborhoods," *TKDE*, vol. 20, no. 1, pp. 55–67, 2008.

[10] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *JMLR*, vol. 7, pp. 2399–2434, 2006.

[11] W. Gao, L. Wang, Y. Li, and Z. Zhou, "Risk minimization in the presence of label noise." in *AAAI*, 2016, pp. 1575–1581.

[12] G. Patrini, F. Nielsen, R. Nock, and M. Carioni, "Loss factorization, weakly supervised learning and label noise robustness," in *ICML*, 2016, pp. 708–717.

[13] T. Liu and D. Tao, "Classification with noisy labels by importance reweighting," *TPAMI*, vol. 38, no. 3, pp. 447–461, 2016.

[14] R. Wang, T. Liu, and D. Tao, "Multiclass learning with partially corrupted labels," *TNNLS*, 2017.

[15] N. Natarajan, I. Dhillon, P. Ravikumar, and A. Tewari, "Learning with noisy labels," in *NIPS*, 2013, pp. 1196–1204.

[16] B. Rooyen, A. Menon, and R. Williamson, "Learning with symmetric label noise: The importance of being unhinged," in *NIPS*, 2015, pp. 10–18.

[17] B. Han, I. Tsang, and L. Chen, "On the convergence of a family of robust losses for stochastic gradient descent," in *ECML-PKDD*. Springer, 2016, pp. 665–680.

[18] Y. Wang, J. Sharpnack, A. Smola, and R. Tibshirani, "Trend filtering on graphs," *JMLR*, vol. 17, no. 105, pp. 1–41, 2016.

[19] C. Gong, D. Tao, K. Fu, and J. Yang, "Fick's law assisted propagation for semisupervised learning," *TNNLS*, vol. 26, no. 9, pp. 2148–2162, 2015.

[20] C. Gong, T. Liu, D. Tao, K. Fu, E. Tu, and J. Yang, "Deformed graph Laplacian for semisupervised learning," *TNNLS*, vol. 26, no. 10, pp. 2261–2274, 2015.

[21] R. Fergus, Y. Weiss, and A. Torralba, "Semi-supervised learning in gigantic image collections," in *NIPS*, 2009, pp. 522–530.

[22] X. Chang, Y. Yu, Y. Yang, and A. Hauptmann, "Searching persuasively: Joint event detection and evidence recounting with limited supervision," in *ACM MM*, 2015, pp. 581–590.

[23] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *TPAMI*, vol. 35, no. 1, pp. 171–184, 2013.

[24] J. Eckstein and D. Bertsekas, "On the douglas-rachford splitting method and the proximal point algorithm for maximal monotone operators," *Mathematical Programming*, vol. 55, no. 1, pp. 293–318, 1992.

[25] S. Nene, S. Nayar, and H. Murase, "Columbia object image library (coil-20)," 1996.

[26] Z. Lu, X. Gao, L. Wang, J. Wen, and S. Huang, "Noise-robust semi-supervised learning by large-scale sparse coding," in *AAAI*, 2015, pp. 2828–2834.

[27] N. Gu, M. Fan, and D. Meng, "Robust semi-supervised classification for noisy labels based on self-paced learning," *IEEE Signal Processing Letters*, vol. 23, no. 12, pp. 1806–1810, 2016.

[28] D. Cai and X. He, "Manifold adaptive experimental design for text categorization," *TKDE*, vol. 24, no. 4, pp. 707–719, 2012.

[29] D. Lewis, Y. Yang, T. Rose, and F. Li, "RCV1: A new benchmark collection for text categorization research," *JMLR*, vol. 5, pp. 361–397, 2004.

[30] X. Chang, Y. Yu, Y. Yang, and E. Xing, "Semantic pooling for complex event analysis in untrimmed videos," *TPAMI*, vol. 39, no. 8, pp. 1617–1632, 2017.