Robust Structure-aware Semi-supervised Learning

Xu Chen
Zoom Video Communications, Inc
steven.xu.chen@gmail.com

2022 IEEE International Conference on Data Mining (ICDM) | 978-1-6654-5099-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICDM54844.2022.00014

***Abstract*—We present a novel unified framework robust structure-aware semi-supervised learning called Unified RSSL (URSSL) which is robust to both outliers and noisy labels where the ground truth of the corrupted labels can be either within or out of training sets. Particularly, URSSL applies joint semi-supervised dimensionality reduction with robust estimators and network sparse regularization simultaneously on the graph Laplacian matrix iteratively to preserve the intrinsic graph structure and ensure robustness to the compound noise. First, in order to relieve the influence from outliers, a novel semi-supervised robust dimensionality reduction is applied relying on robust estimators to suppress outliers. Meanwhile, to tackle noisy labels, the denoised graph similarity information is encoded into the network regularization. Moreover, by identifying strong relevance of dimensionality reduction and network regularization in the context of robust semi-supervised learning (RSSL), a two-step alternative optimization is derived to compute optimal solutions with guaranteed convergence. Extensive experimental results demonstrate the promising performance of this framework when applied to multiple benchmark datasets with respect to state-of-the-art approaches for important applications in the areas of image classification.**

***Index Terms*—regularization, robust estimator, overfitting, embedding, joint optimization.**

I. Introduction

Semi-supervised learning [1][2] has been extensively studied and applied as it seeks to alleviate the need for additional labeled data by incorporating information from unlabeled data in a machine learning model. Due to the subjective nature of manual labeling, human fatigue and the difficulty of the labelling tasks, labels obtained from crowdsourcing (Amazon Mechanical Turk), synthetic labeling and data programming inevitably contain noise. Faced with noisy labels, noise-robust semi-supervised learning algorithms (RSSL) [3][4], which attempt to reduce the detrimental impact of noise in semi-supervised learning, have spurred extensive research interest. However, in practical applications, the noise widely exists not only in the labels but also in the data as sample outliers. For instance, in face recognition, outlier images can come from occlusion and motion blur caused by camera jittering. In medical imaging, deviations in neuroimaging data is due to radiation or patient movements during imaging process.

L_1 norm based sparse regularization serves as one of the most popular robust learning methods due to its effectiveness in denoising noisy labels. Among the popular L_1 norm based robust semi-supervised learning (RSSL) methods [3], two major components including dimensionality reduction and sparse regularization are usually applied. Specifically, dimensionality reduction is first applied to the graph Laplacian matrix based on eigenvector decomposition and subsequently sparse regularization is applied to the reduced eigenvector spaces to alleviate the influence of noisy labels. However, due to the existence of sample outliers, directly computing the graph Laplacian matrix from the raw data results in inaccurate similarity measurement. The work in [5] presents an interesting approach to robust graph dimensionality reduction (RGDR) relying on robust estimators [6] and manifold learning. While RGDR works well in an unsupervised setting, it is tempting to exploit the class-specific information to boost the performance of manifold learning in a semi-supervised manner. Another active area that also focuses on addressing noisy labels is deep learning based approaches [7][8]. However, those approaches are typically requiring sufficient number of accurate training data. While the dimensionality reduction based RSSL methods can usually be training-free, one important gap between the existing robust semi-supervised learning dimensionality reduction approaches [9][10] and deep neural network is that they do not explore the rich information from the network structure in the denoising optimization and therefore frequently result in sub-optimal denoising performance. While the noisy labels and outliers could impact the semi-supervised learning in different ways, they share one common attribute: the corresponding corrupted sample tends to stay far away from intrinsic underlying graph structure guided by label information. Inspired by the fact that network regularization [11] is capable of exploiting network information to remove any redundancies among the features which contains outliers, it is therefore desirable to jointly apply manifold learning and network sparse regularization to characterize the underlying graph structure in a semi-supervised manner to cater for classification of dataset with complicated noise.

Motivated by the aforementioned challenges, we propose a novel unified framework for scalable robust semi-supervised learning by countering both sample outliers and label noise in a single shot called Unified Robust Semi-supervised Learning (URSSL). In particular, we integrate a novel semi-supervised robust graph dimensionality reduction (SRGDR) with network regularization to suppress the noisy labels and sample outliers simultaneously. Unlike the previous loss correction approach, network sparse regularization is applied to group highly correlated samples and pull away noisy labels and sample outliers. By bridging the gap between robust deep neural network with memorization and robust graph-based learning, URSSL neither requires the estimation of the noise transition matrix nor does it suffer from overfitting with high noise ratio. Inherently, a robust embedding constructed

978-1-6654-5099-7/22/$31.00 ©2022 IEEE
DOI 10.1109/ICDM54844.2022.00014

by SRGDR facilitates the efficiency of sparse regularization. The improved sparse regularization in turn further reinforces the SRGDR driven by the cleaned label information. We further empower our framework with a two-step alternative optimization with guaranteed convergence so that the superior classification performance in the presence of noisy datasets over the state-of-the-art approaches are ensured. The major contributions of this paper are summarized as follows :

- By jointly applying a novel semi-supervised robust dimensionality reduction and network graph regularization on the graph Laplacian matrix, the proposed method is capable of calculating the similarity matrix more accurately given the data outlier and suppressing corrupted labels via sparse regularization simultaneously.
- URSSL can tackle the noisy labels with both closed-set and open-set noise via three levels of denoising:
  1) apply SRGDR to capture the intrinsic lower dimensional space to remove the outliers and noise on the input data by employing robust estimator on the graph Laplacian matrix to alleviate the detrimental effect from noisy input data.
  2) leverage a novel network graph regularization to group highly correlated samples and suppress noisy labels.
  3) The optimization problem is solved using a two-step alternative optimization in order to enable the dimensionality reduction and sparse regularization to benefit from each other.
- The convergence of the alternative optimization is proved so that the superior performance over the STOA approaches is guaranteed theoretically.

II. Related Work

RSSL methods have been widely studied and they generally cover two categories including robustness to either sample outliers or noisy labels, but not both together. Towards the challenging issue of label noise, some research work has been done to alleviate the performance degradation due to the incorrect labels in semi-supervised learning. The work in [12] introduces a propagation algorithm that more reliably minimizes a cost function over both a function on the graph and a binary label matrix. Prototype vector machines [2] utilized prototype vectors for efficient approximation on both the graph-based regularizer and model representation. Adaptive neighborhood propagation [13] integrates sparse coding and neighborhood propagation into a unified framework. Importance reweighting has been considered for classification of noise labels in SSL [14]. Gao et al [15] and Patrini et al [16] focus on decomposing the existing loss function as a label-independent term plus a label-dependent term to suppress the negative influence of noisy labels. Other work strived to modify the loss functions for noisy labels by various solutions. Graph trend filtering (GTF)[17] introduced a family of adaptive estimators on the graph by penalizing the L_1 norm of discrete graph differences.

More recently, semi-supervised learning under inadequate and incorrect supervision (SIIS) [4] first applies GTF and then leverages the eigenvectors of the graph Laplacian matrix corresponding to the smallest eigenvalues to reflect the real underlying smoothness of labels for corrupted labels. Regarding the robustness to sample outliers, RFS-LDA [10] employs the least-squares formulation of linear discriminant analysis to detect sample-outliers and feature-noises simultaneously. In [18], reverse graph embedding is applied for identifying the intrinsic structure of the data via dimensionality reduction. In [19], a new family of loss functions called peer loss functions enables learning from noisy labels and does not require a priori specification of the noise rates by working within the standard empirical risk minimization (ERM) framework. The work [20] leverages an intermediate class to avoid directly estimating the noisy class posterior. Despite the great success of these work achieved, frequently the denoising scheme focuses on the processing individual samples or its loss function without consideration of its relation to its neighborhood samples or local graph structure. The connection between the robust deep neural network and the robust graph-based learning has been not well explored. In this work, we combine these two powerful tools in a cohesive way to demonstrate its enormous potential in robust semi-supervised learning.

III. Our Method

A. Preliminaries and Problem Definition

The proposed noise-robust semi-supervised learning algorithm falls into the category of graph-based semi-supervised learning. Assume that we have a point set $\{x_1, \ldots, x_l, x_{l+1}, \ldots, x_n\}$, and the label set L with the number of classes C. The first l data points, $\{x_1, \ldots, x_l\}$, are labeled by $\{y(x_1), \ldots, y(x_l)\}$. The aim is to predict the labels of the unlabeled data points using the information from both the labeled data and the unlabeled data. Namely, we need to find a matrix $F^{n\times C} = [f_1, \ldots, f_C]$ corresponding to the classification of the point set by labeling each x_i on the jth class with the prediction f_{ij}. Further denote the label matrix $Y \in R^{n\times C}$ which contains n vectors $[y_1, \ldots, y_C]$ and for labeled data, $y_{ij} = 1$ if x_i belongs to the jth class. For unlabeled data, y_{ij} is set to be 0. Conventionally, the edge weight between point x_i and x_j, E_{ij}, is calculated by a Gaussian kernel $E_{ij} = \exp(-\|x_i - x_j\|^2/2\sigma^2)$ if $i \neq j$ and $E_{ii} = 0$.

B. Sparse Representation for Semi-supervised Learning

Once we have calulated matrices E, we shall now compute the normalized similarity matrix by $A = D^{-1/2}ED^{-1/2}$, where D is a diagonal matrix with its diagonal element (i, i) equal to the sum of the ith row of A. The normalized Laplacian matrix L can be computed from the weight matrix A: $L = I - A$, where I is a $n \times n$ identity matrix. Subsequently, as L is a positive semi-definite matrix, a symmetrical decomposition is conducted on L. Denote Σ as a diagonal matrix whose diagonal elements as the eigenvalues from L and V as a eigenvector matrix containing the eigenvectors from L. Particularly, L can be decomposed into $L = V\Sigma V^T = (\Sigma^{1/2}V^T)^T\Sigma^{1/2}V^T = B^TB$, here $B = \Sigma^{1/2}V^T$, which is

calculated with all the eigenvectors of the graph Laplacian L, capturing the information relying on the manifold structure of the data. In [3], the L_1-norm is effective for label noise reduction in RSSL

$$\min_F J(F) = \frac{1}{2} \parallel F - Y \parallel_{fro}^2 + \lambda_1 \parallel BF \parallel_1 \quad , \tag{1}$$

The first term of $J(F)$ is the fitting constraint and fro represents the Frobenius norm of a matrix, namely it ensures the classification function to be close enough to the initial classification labels. The second term provides the L_1-norm smoothness constraint for Laplacian regularization and λ_1 controls the trade-off between two terms. In (1), the L_1 norm formulation of Laplacian regularization serves as a critical role in the explanation of noise-robust semi-supervised learning as the aspect of sparse coding. Similar as [3], the above optimization problem can be decomposed in C independent optimization subproblems where each of them cast as:

$$\frac{1}{2} \parallel F_{\cdot k} - Y_{\cdot k} \parallel_{fro}^2 + \lambda_1 \parallel BF_{\cdot k} \parallel_1 \quad , \tag{2}$$

where $F_{\cdot k}$ and $Y_{\cdot k}$ stand for the kth column of F and Y respectively. Denote $F_{\cdot k}, Y_{\cdot k}$ as f_k, y_k. Without loss of generality, for the kth subproblem, the optimization is given by $\min_{f_k} J(f_k) = \frac{1}{2} \parallel f_k - y_k \parallel_2^2 + \lambda_1 \parallel Bf_k \parallel_1$, the dimensionality of $f_k^{n\times 1}$ is reduced by applying eigenvector decompositions of $f_k = V_m \alpha_k$ where V_m is an $n \times m$ matrix where the columns are spanned by m eigenvectors corresponding to the smallest eigenvalues and sparse coefficients. α_k is an $m \times 1$ vector for the kth class. Applying the dimensionality reduction and substituting $B = \Sigma^{\frac{1}{2}} V^T$ and $f_k = V_m \alpha^k$ into the equation (1):

$$J(\alpha_k) = \frac{1}{2} \|(V_m \alpha_k) - y_k\|_2^2 + \lambda_1 \|(\Sigma^{\frac{1}{2}} V^T)(V_m \alpha_k)\|_1 \ . \tag{3}$$

Taking advantage of the orthonormality of eigenvectors in SVD, denote the ith eigenvector as $V_{\cdot i}$, $J(\alpha_k)$ can be simplified as:$\frac{1}{2}\|(V_m \alpha_k) - y_k\|_2^2 + \lambda_1 \| \sum_{i=1}^m \Sigma^{\frac{1}{2}} (V^T V_{\cdot i}) \alpha_{i,k} \|_1$ Essentially, the dimensionality reduction technique is applied in order to guarantee that f is as smooth as possible to reflect the class separation. However, faced with the compound noise which are presented in both of the input data and labels in RSSL, directly applying LSSC could lead to several issues: First of all, the graph Laplacian matrix directly calculated from raw data including outliers is inaccurate. Secondly, the regular eigenvector decomposition did not consider the influence of the outliers thus the reduced space may not represent the true intrinsic low dimensional space, which is critical for suppressing noisy labels. Moreover, the L_1 regularization did not fully explore the correlation between samples and therefore can be further improved by grouping the highly correlated samples with network sparse regularization. Motivated by that, the robust dimensionality reduction method is employed.

C. Robust Dimensionality Reduction

To address the challenging issues of learning with noise labels and features, here a novel semi-supervised robust dimensionality reduction is developed to tackle the noisy distributions and corrupted labels. Define $Z \in R^{m\times n}$ spanned by vectors z_i as the intrinsic structure of the data via the reverse graph embedding and $U \in R^{n\times m}$ as the transformation matrix where $m(m \leq n)$ is the intrinsic space of the original high dimensional data. As the dimensionalty reduction process enables to elimiate the feature dimensions corrupted by outliers, most of the noise is removed through the process of refining the original data to be UZ and dynamically updating the graph matrix S learnt from the intrinsic space of the original data using the following formulation [21]:

$$\min_{U, U^T U = I} \sum_{i,j=1}^{n} s_{i,j} \parallel Uz_i - Uz_j \parallel_2^2 \ . \tag{4}$$

In the context of RSSL, our semi-supervised dimensionality reduction is applied to the graph Laplacian matrix by integrating the label information with robust estimators. In particular, denote Σ_m as the diagonal matrix contains the m smallest eigenvalues from eigenvector decomposition of L. The graph Laplacian matrix based on raw data with dimensionality reduction is computed as

$$L_m = V_m \Sigma_m V_m^T \ . \tag{5}$$

D. Semi-supervised Robust Dimensionality Reduction

We define the robust estimators as the solutions for the eigenvector decomposition of the graph Laplacian matrix based on our semi-supervised robust graph dimensionality reduction algorithm which is insensitive to label noise and outliers. Throughout the paper, the superscript r is used to represent the robust estimator of the corresponding variable. Further denote the robust estimator of L_m as L_m^r. Specifically, given L_m, the goal is to learn a reverse graph embedding V_m^r, a transformation matrix Z such that $L_m^r = V_m^r Z$ which is a good approximation of L_m and a similarity matrix S computed on top of L_m^r under the fitting constraints guided by the labeled data y. The semi-supervised robust graph dimensionality reduction (SRGDR) is proposed as:

$$\begin{gathered} \min_{S, V_m^r, Z} P(S, V_m^r, Z) = \sum_{k=1}^{C} \psi_1(\|(V_m^r \alpha_k) - y_k\|_2) + \\ \psi_2(\parallel L_m - V_m^r Z \parallel_F) + \lambda_3 \sum_{i,j=1}^{n} s_{i,j} \psi_3(\parallel V_m^r z_i - V_m^r z_j \parallel_2) \\ + \lambda_4 \sum_{i=1}^{n} \parallel s_i \parallel_2^2 \quad , \\ \text{s.t. } (V_m^r)^T V_m^r = I, \ \sum_{i=1}^{n} s_i^T \mathbf{1} = 1 \end{gathered}$$

C is the number of classes, ψ_1, ψ_2 and ψ_3 are predefined robust estimators [6], λ_3, λ_4 are positive hyperparameters. The first term guarantees the fitness with the noisy labels relying on robust estimators. The second term serves as the minimization of the error from dimensionality reduction. The third term aims at keeping the similarity between two samples if each of them is one of the k-nearest neighbors of the other. Specifically, it follows the reasonable assumption that if two samples are close enough to each other in high dimensional space, the L_2 norm of the distance for their corresponding embeddings in the lower dimensional space is also small. The fourth term avoids trivial solutions. In particular, by constraining

the norm of s_i, it prevents the exploding coefficients for s_i. The rest of constraints on s_i guarantee different samples have different nearest neighbors. $\mathbf{1}$ is a column vector where all the elements are 1 and $\sum_{i=1}^{n} s_i^T \mathbf{1} = 1$ preserves the shift invariant similarity, where the details of shift invariant similarity can be found at [22]. L_m is precomputed based on raw data using the equation (5) and α is calculated from network sparse regularization introduced next. Unlike RGDR [5] which is unsupervised learning, SRGDR is driven by the labeled data and minimizes the residual between $V_m^r \alpha_k$ and y_k using robust estimator which is expected to generate better performance. Moreover, RGDR is applied only for dimensionality reduction while SRGDR is coupled with sparse regularization and the parameters are jointly learnt in RSSL.

To simplify SRGDR, let x be the union of all x_i. Mathematically, directly optimizing over ψ_1, ψ_2, ψ_3 is difficult. In order to overcome the challenges in optimization, the conjugate function is leveraged. Define $\phi(x)$ as the conjugate function of $\psi(x)$, which can be either an explicit or implicit function. Similar as [23], the optimization of the robust estimator can be transformed into

$$\min_x \sum_{i=1}^{n} \psi(x_i) \Rightarrow \min_{x,p} \sum_{i=1}^{n} p x_i^2 + \phi(p) \tag{6}$$

The core idea is to reparameterize the function $\psi(x_i)$ so that after using the conjugate function, so that the first part of the above formulation $\sum_{i=1}^{n} p x_i^2$ is differentiable over x_i and the second part can be further set as implicit function. The equation (6) indicates that the optimization of $\psi(x_i)$ is equivalent to the optimization of $p x_i^2$ and $\phi(p)$ where p is an auxiliary variable. Further define the minimization function $\delta(x)$ as the function which optimizes $\phi(p)$. Here the following robust estimator and the corresponding minimization function is chosen due to the simplicity:

$$\psi(x) = \frac{x^2}{2(1+x^2)}, \delta(x) = \frac{1}{(1+x^2)^2} \ . \tag{7}$$

We initialize V_m^r and Σ_m^r from V_m and Σ_m by applying SVD on L_m. By minimizing $P(S, V_m^r, Z)$ iteratively, the robust estimators V_m^r, Z are calculated, and then $\Sigma_m^r = Z V_m^r$.

E. Network Constraint Regularization

Define L^r as the robust estimator of the normalized graph Laplacian matrix L and $L_m^r \in m \times m$ as the robust estimator of the normalized graph Laplacian after reducing to m dimensions constructed with the smallest m eigenvalues. Further denote Σ^r as the robust diagonal matrix obtained from the eigenvector decomposition of L^r. To model the graph dependency among the subclasses from image datasets in the presence of noisy input, we formulate the optimization of the sparse coefficients α with robust network constraint regularization defined as the third term in the following optimization as

$$\begin{aligned} Q(\alpha_k) = \tfrac{1}{2}\|(V_m^r \alpha_k) - y_k\|_2^2 + \lambda_1 \|(\Sigma^{\frac{1}{2}} V^T)(V_m^r \alpha_k)\|_1 \\ + \lambda_2 \alpha_k^T L_m^r \alpha_k \ . \end{aligned} \tag{8}$$

Here the third term not only captures the graph structure of images where here it refers to the graph Laplacian matrix L_m^r but also naturally integrates the robustness of the network regularization algorithm by finding the intrinsic embedding given the noisy labels coming from mislabeling and outliers generated from object occlusions, blurred images due to fast motions and out of focus, etc. We identified that if we rewrite equation 8 as a variant form of sparse coding by reparameterization so that the fast iterative shrinkage-thresholding algorithm (FISTA) algorithm can be applied to reduce the computational burden and ensure the linear time complexity.This can be achieved by matrix augmentation. Therefore, we augment the variables $V_m^{r*} \in R^{(n+m)\times m}, V^{r*} \in R^{(n+m)\times n}$ with the following transformation,

$$V_m^{r*} = (1+\lambda_2)^{-\frac{1}{2}} \begin{pmatrix} V_m^r \\ \sqrt{\lambda_2} \Sigma_m^{\frac{1}{2}} V_m^{rT} \end{pmatrix} \tag{9}$$

$$V^{r*} = (1+\lambda_2)^{-\frac{1}{2}} \begin{pmatrix} V^r \\ \sqrt{\lambda_2} \Sigma_m^{\frac{1}{2}} V^{rT} \end{pmatrix},$$

Further augment $y_k^* \in R^{(n+m)\times 1} = [y_k, 0]$. The optimization cost function can then be rewritten as

$$Q(\alpha_k) = \frac{1}{2}\|(V_m^{r*} \alpha_k) - y_k^*\|_2^2 + \lambda_1 \|(\Sigma^{\frac{1}{2}} V^{r*T})(V_m^{r*} \alpha_k)\|_1 \ . \tag{10}$$

This reparameterization not only facilitates the joint learning for batch processing but also paves the way for learning with streaming data using recursive solutions which will be discussed later. Similarly as (2), $Q(\alpha_k)$ can be simplified as:

$$\begin{aligned} &\tfrac{1}{2}\|(V_m^{r*} \alpha_k) - y_k^*\|_2^2 + \lambda_1 \| \textstyle\sum_{i=1}^{m} (\Sigma_{ii}^r)^{\frac{1}{2}} ((V^r)^{*T} V_{.i}^{r*}) \alpha_{i,k} \|_1 \\ &= \tfrac{1}{2}\|(V_m^{r*} \alpha_k) - y_k^*\|_2^2 + \lambda_1/(1+\lambda_2) \textstyle\sum_{i=1}^{m} (\Sigma_{ii}^r)^{\frac{1}{2}} |\alpha_{i,k}| \ . \end{aligned}$$

F. Joint Learning

Denote the robust truncated diagonal matrix with the m smallest eigenvalues as $\Sigma_m^r \in R^{m\times m}$, noticing that $L_m^r = V_m^r Z = V_m^r \Sigma_m^r (V_m^r)^T$, we obtain $Z = \Sigma_m^r (V_m^r)^T$. In order to combine $Q(\alpha_k)$ and $P(S, V_m^r, Z)$, substitute Z with $\Sigma_m^r (V_m^r)^T$ in $P(S, V_m^r, Z)$ and replace the robust estimator ψ with the conjugate function ϕ, the proposed full URSSL optimization problem is formulated as:

$$\begin{aligned} \min J(S, V_m^r, \Sigma_m^r, \alpha, e, b, c) = \textstyle\sum_{k=1}^{C} e_k \|(V_m^{r*} \alpha_k) - y_k^*\|_2^2 + \\ \textstyle\sum_{i=1}^{n} b_i \parallel L_m - V_m^r \Sigma_m^r (V_m^r)^T \parallel_2^2 + \\ \lambda_3 \textstyle\sum_{i,j=1}^{n} s_{i,j} c_{i,j} \parallel V_m^r (\Sigma_m^r (V_m^r)^T)_i - V_m^r (\Sigma_m^r (V_m^r)^T)_j \parallel_2^2 \\ + \phi(e) + \phi(b) + \phi(c) + \lambda_4 \textstyle\sum_{i=1}^{n} \parallel s_i \parallel_2^2 + \frac{\lambda_1}{1+\lambda_2} \\ \textstyle\sum_{k=1}^{C} \sum_{i=1}^{m} (\Sigma_{ii}^r)^{\frac{1}{2}} \mid \alpha_{i,k} \mid \ , \text{s.t. } (V_m^r)^T V_m^r = I, \\ \textstyle\sum_{i=1}^{n} s_i^T \mathbf{1} = 1 \end{aligned} \tag{11}$$

where $(\Sigma_m^r (V_m^r)^T)_i$ represents the ith column of the estimated matrix Z. $e \in R^{C\times 1}$, $b \in R^{n\times 1}$ and $c^{n\times n}$ are two auxiliary vectors and an auxiliary matrix respectively. URSSL focuses on directly minimizing the residual of the graph Laplacian and relying on the estimated robust eigenvectors V_m^r and

eigenvalues Σ_m^r in the optimization. A two-step alternative optimization is proposed. First, we fix $S, V_m^r, e, b, c, \Sigma_m^r$ and optimize α by solving C independent minimization problems $e_1 J_1(\alpha_1), \ldots, e_k J_1(\alpha_k), \ldots, e_C J_1(\alpha_C)$. The optimization of the kth problem is equivalent to

$$J_1(\alpha_k) = e_k \parallel V_m^{r*}\alpha_k - y_k^* \parallel_2^2 + \frac{\lambda_1}{1+\lambda_2} \sum_{i=1}^{m} (\Sigma_{ii}^r)^{\frac{1}{2}} \mid \alpha_{i,k},|$$

G. Optimization and Speedup

To ensure the linear time and space complexity of the proposed algorithm with respect to n, here the Fast Iterative Shrinkage Thresholding Algorithm (FISTA) [24] is employed. In order to apply FISTA to minimize $J_1(\alpha_k)$, rearrange $J_1(\alpha_k)$ to be the summation of a differentiable function and a non-differentiable function: $J_1(\alpha_k) = C_1(\alpha_k) + C_2(\alpha_k)$, where $C_1(\alpha_k) = e_k \|(V_m^{r*}\alpha_k) - y_k^*\|_2^2$ and $C_2(\alpha_k) = \frac{\lambda_1}{1+\lambda_2} \sum_{i=1}^{m} (\Sigma_{ii}^r)^{\frac{1}{2}} |\alpha_{i,k}|$. The gradient of $C_1(\alpha_k)$ is

$$\nabla_{\alpha_{i,k}}(C_1(\alpha_{i,k})) = C_{3,i} \tag{12}$$

where $C_3^{m\times 1} = 2e_k(V_m^r)^{*T}(V_m^{r*}\alpha_k - y_k^*)$ and $C_{3,i}$ represents the ith element of C_3. The soft-thresholding function in FISTA $\Delta_{i,k}$ is:

$$\Delta_{i,k} = \alpha_{i,k} - \nabla_{\alpha_{i,k}}(C_1(\alpha_{i,k}))/\|V_m^{r*}\|_s^2 \ . \tag{13}$$

where $\|V_m^{r*}\|_s$ characterizes the spectral norm of the matrix V_m^{r*}. If $\Delta_{i,k} > 0$, $\alpha_{i,k} = \max\{\Delta_{i,k} - \frac{\lambda_1}{1+\lambda_2}(\Sigma_{ii}^r)^{1/2}/\|V_m^{r*}\|_s, 0\}$. If $\Delta_{i,k} \leq 0$, $\alpha_{i,k} = \min\{-\Delta_{i,k} - \frac{\lambda_1}{1+\lambda_2}(\Sigma_{ii}^r)^{1/2}/\|V_m^{r*}\|_s, 0\}$. Once the optimized α is computed, we then fix α to optimize for $S, V_m^r, e, b, c, \Sigma_m^r$ on J Similarly as [25], the results of e, b, c are related to the residuals of $(V_m^{r*}\alpha_k - y_k^*)$, $(L_m - V_m^r\Sigma_m^r(V_m^r)^T)$ and $(V_m^r(\Sigma_m^r(V_m^r)^T)_i - V_m^r(\Sigma_m^r(V_m^r)^T)_j)$. When the residuals are large, the weights will be small. By using this formulation, the sample outliers are suppressed or removed. Different from J, as the terms in J_2 are all differentiable, the optimization of J_2 is relying on the gradient descent algorithm, where the gradient is calculated via each variable respectively when other variables are fixed. We iteratively optimize $J_1(\alpha)$ and J_2 until all the statistics are stabilized. The joint optimization is initialized with setting V_m^r, Σ_m^r as identity matrices and setting α_k as identity vectors. Finally, denote the optimized classification function and sparse coefficients as f_k^{op} and α_k^{op}, $f_k^{op} = V_m^r * \alpha_k^{op}$. Denote $f(i)_j^{op}$ as the ith element for the optimized classification function for the jth class, a sample x_i is classified into a class that satisfies $\arg\max_{1\leq j\leq C} f(i)_j^{op}$. In addition, when the auxiliary variables e, b, c reduce to the identity vectors and the identity matrix, the solution boils down to solving the semi-supervised learning problem with the clean data distributions.

As deep neural network is very popular for imaging tasks due to their scalability and convenience in the training and deployment and feature representation plays an important role in improving the classification performance, our framework can be easily coupled with deep neural network by extracting features from the output of the last hidden layers (before softmax activations).

TABLE I
DATA STATISTICS ON EVALUATED BENCHMARK DATASETS.

Dataset	Class Numbers	Sample Numbers
MNIST	10	60000
CIFAR-10	10	60000
CIFAR-100	100	60000
Clothing1M	14	1M
WebVision	1000	2.4M

IV. EXPERIMENTS

A. Image Classification

1) Dataset and Competing Methods: We evaluate the robustness and accuracy performance of the proposed algorithm by comparing with multiple baseline methods and the state-of-the-art approaches. The baseline methods consist of meanS3VM [26] ROSSEL [1] and RGDR+LSSC [5]. The state-of-art approaches include Dividemix [27], Peer loss functions [19], DualT [20], M-correction [8], P-correction [7], Meta-Learning [28], Coteaching [29], RFS-LDA [10], Class-sharing data detection and feature adaptation (CAFA) [30] and DP-SSL[31]. Among them, ROSSEL generates pseudo-labels for unlabeled data using 50 weak annotators in SSL. RGDR+LSSC works by first applying RGDR to the graph Laplacian matrix and then conduct L_1 norm based regularization for denoising and classification. Meta-Learning [28] applies a noise-tolerant training algorithm relying on a meta-learning update. P-correction [7] tackles the noisy labels by training an end-to-end framework which can update network parameters and label estimations as label distributions. Iterative-CV [32] applies cross-validation to randomly split noisy datasets and adopts Coteaching[29] techniques to train DNNs robustly against noisy labels. Dividemix [27] models the per-sample loss with a mixture model to dynamically divide the training data into a labeled set with clean samples and an unlabeled set with noisy samples and trains two diverged networks simultaneously. CAFA[30] targets on a more general case where a mismatch exists in both class and feature distribution. DP-SSL that employs an innovative data programming (DP) scheme to generate probabilistic labels for unlabeled data.

We evaluate the performance on five benchmark datasets including the MNIST, CIFAR-10, CIFAR-100, Clothing1M and WebVision datasets for classification accuracy. The proposed algorithm is also evaluated with two real world large scale image datasets including Clothing1M and WebVision1.0. For Clothing1M dataset, it includes 1 million training images obtained from online shopping websites and labels are generated from surrounding texts. WebVision includes 2.4 million images collected from the internet using the 1,000 concepts in ImageNet ILSVRC12. Similar to the previous work [32], the baseline methods on the first 50 classes of the Google image subset using the inception-resnet v2 [33] are compared. See Table I for the data statistics on evaluated benchmark datasets.

2) Implementation Details: In order to extract discriminative features on image data, for MNIST, CIFAR-10 and CIFAR-100 datasets, a PreAct ResNet-32 [34] trained with SGD using a batch size of 128 is employed for fair comparison with existing methods with a momentum of 0.9, a weight decay of 0.0005 and the output from the last hidden layer of ResNet is used to generate the corresponding feature vectors which can then be utilized as input for the URSSL algorithm. The PreAct Resnet-32 network is trained for 300 epochs and the initial learning rate is set to be 0.02. For Clothing1M dataset, following the previous work [28], a ResNet-50 with ImageNet pretrained weights are utilized and the embeding from the last hidden layer of Resnet-50 are extracted. For both of Clothing1M and WebVision datasets, the network is trained using SGD with a momentum of 0.9, a weight decay of 0.001, and a batch size of 32. The Clothing1M is trained with 80 epochs. The initial learning rate is set as 0.002 and reduced by a factor of 10 after training with 40 epochs. Within each epoch, 1000 mini-batches are sampled from the training data and the balance of labels (noisy) are ensured.

We set the stop criteria of URSSL as $\frac{|obj(t+1)-obj(t)|_2^2}{obj(t)} \leq 10^{-5}$ where obj is the objective function value J for the joint optimization. Typically, Both CIFAR-10 and CIFAR-100 contain 50K training images and 10K test images of size 32 by 32. The hyperparameter ranges for the URSSL algorithm are $\sigma = [0.001; 10000]$, $\lambda_1, \lambda_2, \lambda_3, \lambda_4 = [10^{-2}, 10^2]$, the number of clusters $T = [1000, 5000]$, the number of nearest neighbors for low rank approximations $r = [5, 10]$. The grid search for model hyperparameter tuning is applied and the best performance is reported for each method. The reduced dimension m is selected to be 20 percent of the total number of samples. Each dataset is randomly sampled and divided into three disjointed subsets including the labeled set (5% samples), unlabeled set (75% samples) and test set (20% samples). We include 10% sample outliers to all the datasets where the outliers are created by randomly removing 20% features in the data and replacing with zeros. The random sampling is conducted over 20 runs and the average classification accuracies are reported. Two types of label noises are studied in the experiments including symmetric and asymmetric label noise. In particular, the symmetric noise is ranging from 20% to 80% and generated by randomly replacing the labels for a percentage of training data with all possible labels. The asymmetric noise is created to simulate the real-world label noise where the corrupted label consists of the labels from the most similar class with respect to the ground truth (e.g. "horse" to "deer", "truck" to "automobile", "bird" to "airplane").

3) Qualitative Evaluations: Fig.1 demonstrates the comparison of the t-SNE visualizations to two dimensional spaces for CIFAR-10 dataset by applying DivideMix [27] and our URSSL on the extracted deep learned features with sample outliers and noisy labels, where the index number indicates classes 0 to 9. Shown from Fig.1, our URSSL outperforms DivideMix by providing a more discriminative embedding leveraging the class specific information and joint optimization with sparse regularization. Moreover, URSSL preserve better network structures of the superclasses (such as "fish" and "aquatic mammals"). URSSL provides a dramatically improved effect for denoising the compound noise and boosting the accuracy thanks to the joint optimization to group correlated images from the same class and keep outliers as far from other images as possible and the robust loss function to tackle the compound noise. The strong correlation between the image features are verified by significant p values in the Student's t-distribution test on Pearson correlation coefficients at the 0.05 level under the null hypothesis that there is no linear relationship between two samples. Fig.2 provides exemplary images with noisy labels detected by our method from Clothing1M dataset, where the false labels are above the images in orange and the true labels are below the images in light blue. These examples demonstrates the efficacy of our method in detection of images with noisy labels. We analyzed the successful and failure cases in the experiments and identified that most of the failure cases corresponding to the minor classes where there are not enough samples for learning the specific pattern.

Fig. 1. Comparison of the t-SNE visualization at two dimensional spaces for the CIFAR-100 dataset using DivideMix [27] and our URSSL with 10% sample outliers and 20% noisy labels, where the index number indicates superclasses. Each number locates on the median position of the corresponding vectors and the outliers are marked with squares. The embeddings from 10 distinct clusters using our method corresponds to true superclass labels instead of noisy labels, which justifies the robustness of our method to label noise and sample outliers.

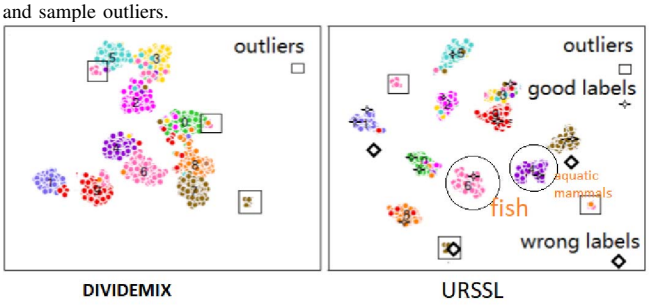
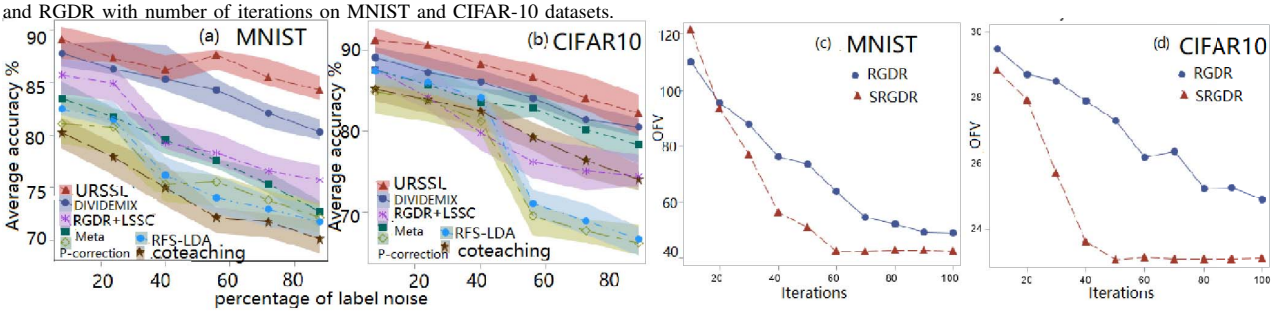


Fig. 2. Exemplary images with noisy labels detected by our method from Clothing1M dataset, where the false labels are above the images in orange and the true labels are below the images in light blue.

Fig. 3. (a-b) Average accuracy (95% confidence intervals) for MNIST and CIFAR-10 datasets over 20 runs with varying noise level for labels where labeled data includes 5% samples. The input data also includes 10% sample outliers.(c-d) The variations of objective function values (OFV) of the proposed SRGDR and RGDR with number of iterations on MNIST and CIFAR-10 datasets.

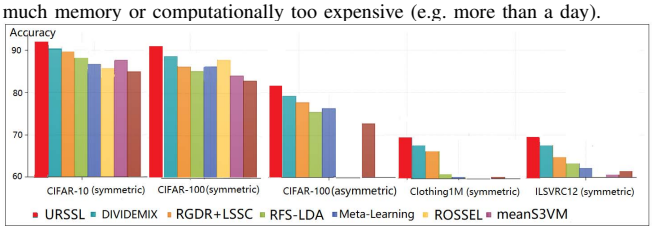


4) Quantitative Evaluations: Shown in Fig.3(a-b), as label noise (the percentage of inaccurately labeled samples among all labeled samples) increases, all the compared methods have performance degradations as expected, where the degradation from URSSL is the minimum, demonstrating the robustness of the proposed algorithm. Typically, the performance gain of URSSL over Dividemix can be mainly attributed to the fact that the URSSL counters the label noise and sample outliers simultaneously and avoids overfitting by using the proposed training free approach in the joint optimization. While the performance of Dividemix and other DNN-based methods deteriorates due to overfitting with high noise ratio. Compared RGDR+LSSC, SRGDR in URSSL finds a better embedding guided by the label information compared to the unsupervised RGDR. The improved embedding enhances the sparse regularization by eliminating redundant data and outliers. Meanwhile, the proposed robust loss function in DNN also suppresses the noisy labels and turns out to lead to better dimensionality reduction. Fig.3(c-d) demonstrates the variations of objective function values (OFV) of the proposed SRGDR and RGDR with number of iterations on MNIST and CIFAR-10 datasets. It verifies that our SRGDR converges much faster by encoding the class specific information and results in a better embedding.

We report the performance of classification accuracy with symmetric label noise by comparing with existing approaches on the CIFAR-10 and CIFAR-100 datasets in Table II, where the best accuracy over all the epochs are reported. We observe the performance margin of our method increases with the noise ratio and outlier percentages, which demonstrates the robustness of the proposed method. The superiority of our URSSL over Dividemix[30] and CAFA[30] can be mainly attributed to the iterative joint learning and denoising algorithm in URSSL towards complicated noise.

5) Hyperparameter Study: Relying on the automatic hyperparameter selection technique in [35] based on a combination of a genetic algorithm (GA) and a generating set search (GSS) technique for searching the optimal hyperparameter, we find that the proposed algorithm is not sensitive to the regularization parameters λ_3 and λ_4. With the CIFAR-10 dataset , the

Fig. 4. Comparison of classification accuracy for seven different algorithms with 5 percent labeled data where 20 percent of labeled data are contaminated by noise. 10 percent of sample outliers are included into the input data on CIFAR-10, CIFAR-100 (symmetric and asymmetric noise), Clothing1M and ILSVRC12 datasets (Note some algorithms fail due to either requires too much memory or computationally too expensive (e.g. more than a day).



performance of URSSL is improved by 1.6% compared to the overall improvement of 5.1% by tuning only the regularization parameter on the L_1 norm λ_1. While for the CIFAR-100 dataset, tuning only λ_2 provides an additional 1.4% gain in classification accuracy. Fig.4 demonstrates the comparison of the classification accuracy on five image datasets using eight different algorithms with 5 percent labeled data where 10 percent of labeled data are noisy. In addition, 10 percent of input data are contaminated by sample outliers. Comparing with RGDR+LSSC, SRDGR alone contributes around 65% performance gain on average with respect to the second best competing method. Specifically, with the most challenging case CIFAR-100 dataset containing 80% label noise and 20% sample outliers, our method significantly outperforms the best competitor Dividemix[27] by 7.5%.

Table III provides the comparison of classification accuracy using different learning algorithms on the CIFAR-10 (with asymmetric 40% label noise and 10% outliers) and Clothing1M datasets (real-world noise) along with standard deviation (in brackets). 40% asymmetric label noise is selected because certain classes become theoretically indistinguishable for asymmetric noise larger than 50%. Joint-Optim [37] jointly optimize the sample labels and the network parameters. As it can be seen from Table III that our regularization works nicely with asymmetric noise and real-world noise. With joint regularization, our method significantly outperform the best competitor Dividemix by 3.8% and 5.2% respectively on CIFAR-10 and Clothing1M datasets for batch processing. For

TABLE II

COMPARISON OF CLASSIFICATION ACCURACY USING DIFFERENT LEARNING ALGORITHMS ON THE CIFAR-10 AND CIFAR-100 DATASETS WITH VARYING LEVELS OF LABEL NOISE AND SAMPLE OUTLIERS ALONG WITH STANDARD DEVIATION (IN BRACKETS). WE RE-IMPLEMENT ALL METHODS UNDER THE SAME SETTING BASED ON PUBLIC CODE.

Dataset	CIFAR-10	CIFAR-10	CIFAR-10	CIFAR-10	CIFAR-100	CIFAR-100	CIFAR-100	CIFAR-100
Label Noise, Outliers	50%, 10%	80%, 10%	50%, 20%	80%,20%	50%, 10%	80%,10%	50%, 20%	80%,20%
SIIS[4]	68.2(0.7)	62.3(3.2)	78.6(0.5)	73.1(0.5)	48.7(0.8)	28.3(0.5)	47.2(0.6)	25.3(0.5)
RPCA+LSSC[36][3]	70.3(0.4)	67.7(0.6)	68.1(0.4)	51.3(0.3)	43.5(0.6)	46.7(0.6)	38.5(0.6)	36.2(0.3)
RGDR+LSSC[5][3]	72.6(0.7)	69.3(0.5)	69.6(0.5)	67.8 (0.6)	53.7(0.5)	46.5(0.6)	52.3(0.6)	41.3(0.5)
Coteaching[29]	85.1(1.3)	65.8(0.5)	81.3(0.6)	61.5(0.4)	53.1(0.8)	21.9(0.7)	51.6(0.5)	20.7(0.6)
M-correction[8]	79.7(0.8)	62.9(0.6)	75.7(0.5)	72.6(0.5)	67.3(0.7)	48.6(0.8)	65.6(0.3)	47.3(0.5)
P-correction[7]	86.3(0.9)	75.1(0.6)	74.5(0.5)	72.3(0.5)	65.3(0.7)	58.3(0.8)	61.6(0.3)	46.9(0.4)
Meta-Learning [28]	87.9(1.5)	84.0(1.6)	75.2(0.6)	72.6(0.4)	59.1(0.6)	46.6(0.7)	58.7(0.4)	45.3(0.5)
Dividemix [27]	91.3(1.6)	90.7(2.0)	90.5 (0.4)	88.7(0.6)	72.1(0.7)	57.9(0.7)	68.2(0.6)	56.8(0.5)
CAFA[30]	91.5(0.6)	91.0(0.7)	90.6(0.5)	89.3(0.3)	72.7(0.4)	58.3(0.5)	69.7(0.5)	57.2(0.3)
DPSSL[31]	91.3(0.5)	90.9(0.8)	90.4(0.7)	89.2(0.4)	72.5(0.5)	58.2(0.4)	69.3(0.2)	57.0(0.4)
URSSL	**95.7(0.6)**	**94.5(0.7)**	**95.3(0.6)**	**94.1(0.5)**	**79.6(0.2)**	**65.6(0.4)**	**76.5(0.4)**	**64.3(0.6)**

TABLE III

COMPARISON OF CLASSIFICATION ACCURACY USING DIFFERENT LEARNING ALGORITHMS ON THE CIFAR-10 (WITH ASYMMETRIC 40% LABEL NOISE AND 10% OUTLIERS) AND CLOTHING1M DATASETS (REAL-WORLD NOISE AND 10% OUTLIERS) ALONG WITH STANDARD DEVIATION (IN BRACKETS).

Dataset	CIFAR10	Clothing1M
AELP-WL[38]	81.6(0.9)	64.72(0.6)
SIIS [4]	83.2(0.7)	62.13(3.2)
RPCA+LSSC	84.5(0.4)	65.37(0.6)
M-correction[8]	87.1(0.5)	70.53(0.4)
Joint-Optim[37]	87.6(1.3)	71.35(0.5)
P-correction[7]	86.3(0.9)	72.81(0.6)
Meta-Learning [28]	87.9(1.5)	73.01(0.7)
Peer loss[19]	90.5(0.8)	73.09(0.8)
DualT [20]	90.2(1.1)	73.03(0.9)
Dividemix [27]	91.3(1.6)	73.16(1.2)
CAFA[30]	90.8(0.5)	73.12(0.4)
DPSSL[31]	90.6(0.3)	73.17(0.6)
URSSL	**95.1(0.6)**	**78.32(0.7)**

recursive processing, our method yields 3.5% and and 4.5% in terms of performance gain. In contrast, most approaches from the competitors cannot address the issues from outliers and label noise simultaneously. Table IV illustrates the comparison of top-1 (top-5) accuracy with different state-of-the-art methods on the WebVision validation dataset and the ImageNet ILSVRC12 validation datasets training on (mini)WebVision dataset. Here top-5 accuracy is an extension to top-1 accuracy where instead of computing the probability that the most probable class label equals to the ground truth label, the probability that the group truth label is in the top 5 most probable labels is calculated. Specifically, in MentorNet[39], an auxiliary teacher network is pre-trained and used to drop samples with noisy labels for its student network. Then, student network is used for image recognition. We report the comparison of classification accuracy for URSSL under the setting of either noisy labels (L) or outliers (O) in Table V, where URSSL consistently demonstrates the superiority over the state-of-the-art approaches.

Here we provides some details on ablation study in Table VI. URSSL w/o SRGDR has the same robust estimators, network sparse regularizations and joint optimization. But the dimensionality reduction is conducted in an unsupervised manner. Therefore the performance drop (especially with 80% label noise) suggests the importance of semi-supervised dimensionality reduction. Secondly, URSSL w/o robust estimators use the same setting as URSSSL except the robust estimators which is important for countering the problem of

TABLE IV

COMPARISON OF TOP-1 (TOP-5) ACCURACY WITH DIFFERENT STATE-OF-THE-ART METHODS ON THE WEBVISION VALIDATION DATASET AND THE IMAGENET ILSVRC12 VALIDATION DATASETS TRAINING ON (MINI)WEBVISION DATASET.

Dataset	WebVision	WebVision	ILSVRC12	ILSVRC12
Metric	top1	top5	top1	top5
Coteaching[29]	62.75	83.61	60.73	83.56
F-correction[40]	60.73	81.64	56.81	81.72
Decoupling[41]	61.37	82.95	57.93	81.38
MentorNet[39]	62.78	80.92	57.52	79.51
Iterative-CV [32]	64.87	84.03	61.31	83.79
Peer loss[19]	73.45	86.33	71.55	85.06
DualT [20]	73.69	85.77	68.53	84.25
Dividemix [27]	75.68	87.73	72.87	85.61
CAFA[30]	75.32	86.85	72.11	84.89
DPSSL[31]	75.27	86.31	72.05	84.43
URSSL	**78.31**	**92.45**	**76.31**	**91.23**

TABLE V

COMPARISON OF THE CLASSIFICATION ACCURACY OF URSSL ON CIFAR10 (C10) AND CIFAR100 (C100) UNDER EITHER NOISY LABELS OR OUTLIERS (E.G. L50 DENOTES NOISY LABEL RATE AS 50%, O20 REPRESENTS 20% OUTLIERS).

Datasets	C10	C10	C10	C10	C100	C100	C100	C100
Noise	L50	L80	O10	O20	L50	L80	O10	O20
Coteach	87.3	68.4	88.2	67.3	67.6	34.3	62.4	43.8
P-correct	86.8	77.3	87.5	79.6	73.5	62.7	67.8	57.1
Meta-Learn	88.7	85.2	88.3	77.9	62.2	59.8	63.0	55.9
DivideMix	92.6	91.0	91.7	89.1	75.7	71.2	74.6	64.3
CAFA	91.7	88.6	91.2	90.6	73.3	70.9	74.2	64.1
URSSL	**95.9**	**94.6**	**94.5**	**93.5**	**83.3**	**79.5**	**85.9**	**76.2**

sample outliers because more outliers would be mistakenly classified without robust estimators. Finally, from the ablation study, it is confirmed that the network sparse regularization indeed strengthened the classification performance. It can also be seen the ablation study, for asymmetric noise, the robust loss function plays the most important role which can mainly be attributed to the fact that the iterative joint refinement of the dimensionality reduction and sparse regularization is more beneficial to the asymmetric label noise.

We analyze the training time (hours) of URSSL to demonstrate its computational efficiency. The training time of URSSL is reported on the CIFAR10 dataset with several state-of-the-art approaches, evaluated on a single Nvidia V100 GPU in Table VII. As expected, URSSL is slower than Coteaching, however is faster than the rest of competing methods due to efficient SRGDR and effective network regularization. In particular, the reduced running time of URSSL compared to DivideMix can be mainly attributed to efficient optimization.

V. Conclusion

In this paper, we presented a novel scalable RSSL approach under noisy labels and sample outliers with the application on large scale image classification and face recognition. With the proof of convergence, we offer the new insight that the robust deep neural network and robust graph-based learning can benefit from each other when applied jointly in URSSL. The proposed approach can be easily integrated into robust deep learning. The suppression of noisy labels and sample outliers is independent of the knowledge of the noise type and estimation of the noise statistics. Evaluations on multiple benchmark and real-world datasets demonstrate the efficiency and robust ness of URSSL compared to the state-of-the-art approaches. Future work will focus on the extension of the current framework to online learning for streaming data.

Appendix A
Proof of Convergence Analysis

Denote $(V_m^r)^t, S^t, (\Sigma_m^r)^t, \alpha^t, e^t, b^t, c^t$ as the variables at the tth iteration, fix $(V_m^r)^{t+1}, S^{t+1}, (\Sigma_m^r)^{t+1}, \alpha^{t+1}, c^{t+1}, b^{t+1}$ to optimize over e, we obtain

$$J((V_m^r)^{t+1}, S^{t+1}, (\Sigma_m^r)^{t+1}, \alpha^{t+1}, c^{t+1}, b^{t+1}, e^{t+1}) \leq J((V_m^r)^{t+1}, S^{t+1}, (\Sigma_m^r)^{t+1}, \alpha^{t+1}, c^{t+1}, b^{t+1}, e^t) \ . \quad (14)$$

Similar inequalities can be shown for the auxiliary variables b and c. Due to the convergence property for FISTA algorithm, fix $(V_m^r)^{t+1}, S^{t+1}, (\Sigma_m^r)^{t+1}, c^t, e^{t+1}, b^{t+1}$, we have

$$J((V_m^r)^{t+1}, S^{t+1}, (\Sigma_m^r)^{t+1}, \alpha^{t+1}, e^{t+1}, b^{t+1}, c^t) \leq J((V_m^r)^{t+1}, S^{t+1}, (\Sigma_m^r)^{t+1}, \alpha^t, e^{t+1}, b^{t+1}, c^t) \ . \quad (15)$$

When fixing other variables, V_m^r is optimized with gradient descent algorithm, where the gradient of V_m^r is calculated as:

$$\nabla_{V_m^r} = \sum_{c=1}^K 2(1+\lambda_2)^{-1/2} e_k \alpha_k \begin{pmatrix} I \\ \sqrt{\lambda_2}\Sigma_m^{\frac{1}{2}} \end{pmatrix} (V_m^{r*}\alpha_k) - y_k^*) + 2\lambda_3 \sum_{i=1}^n s_{i,j} c_{i,j} \Sigma_m^r (V_m^r)^T (V_m^r (\Sigma_m^r (V_m^r)^T)_i - V_m^r (\Sigma_m^r (V_m^r)^T)) + 2\lambda_2 \sum_{i=1}^n b_i \Sigma_m^r (V_m^r)(L_m - V_m^r \Sigma_m^r (V_m^r)^T \quad (16)$$

Meanwhile, Σ_m^r and S can be computed in the closed form solutions. The gradient of Σ_m^r can be represented as:

$$\nabla_{\Sigma_m^r} = -2\sum_{i=1}^n b_i (V_m^r (V_m^r)^T)(L_m - V_m^r \Sigma_m^r (V_m^r)^T) + \lambda_3 \sum_{i,j=1}^n s_{i,j} c_{i,j} (V_m^r (\Sigma_m^r (V_m^r)^T)_i - V_m^r (\Sigma_m^r (V_m^r)^T)_j)(V_m^r ((V_m^r)^T)_i - V_m^r ((V_m^r)^T)_j) \quad (17)$$

By setting the gradient to be zero, the closed form solutions for Σ_m^r can be calculated. Thus, similar inequalities can be derived. In summary, we have

$$J((V_m^r)^{t+1}, S^{t+1}, (\Sigma_m^r)^{t+1}, \alpha^{t+1}, c^{t+1}, b^{t+1}, e^{t+1}) \leq J((V_m^r)^t, S^t, (\Sigma_m^r)^t, \alpha^t, c^t, b^t, e^t) \ . \quad (18)$$

References

[1] Y. Yan, Z. Xu, and I. Tsang, "Robust Semi-Supervised Learning through Lable Aggregation," *AAAI Conference on Artificial Intelligence*, 2016.

[2] K. Zhang, J. Kwok, and B. Parvin, "Prototype vector machine for large scale semi-supervised learning," *International Conference on Machine Learning (ICML)*, 2009.

[3] Z. Lu, X. Gao, L. Wang, J. Wen, and S. Huang, "Noise-Robust Semi-Supervised Learning by Large-Scale Sparse Coding," *AAAI Conference on Artificial Intelligence*, 2015.

[4] C. Cong and H. Zhang, "Learning with Inadequate and Incorrect Supervision," *International Conference on Data Mining (ICDM)*, 2017.

[5] X. Zhu, C. Lei, and H. Yu, "Robust Graph Dimensionality Reduction," *International Joint Conference on Artificial Intelligence*, 2018.

[6] P. Huber, "Robust Statistics," *Interna-tional Encyclopedia of Statistical Science*, 2011.

[7] K. Yi and J. Wu, "Probabilistic end-to-end noise correction for learning with noisy labels," *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[8] E. Arazo, D. Ortego, P. Albert, and N. Connor, "Unsupervised Label Noise Modeling and Loss Correction," *International Conference on Machine Learning*, 2019.

[9] H. Gan, "A noise-robust semi-supervised dimensionality reduction method for face recognition," *International Journal of Lights and Electron Optics*, 2017.

[10] E. Adeli, K. Thung, L. An, F. Shi, and D. Shen, "Semi-Supervised Discriminative Classification Robust to Sample-Outliers and Feature-Noises," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[11] J. Yoon and S. Hwang, "Combined Group and Exclusive Sparsity for Deep Neural Networks," *International Conference on Machine Learning*, 2017.

[12] J. Wang, T. Jebara, and S. Chang, "Graph Transduction via Alternating Minimization," *ICML*, 2008.

[13] L. Jia, Z. Zhang, L. Wang, W. Jiang, and M. Zhao, "Adaptive Neighborhood Propagation by Joint L2,1-Norm Regularized Sparse Coding for Representation and Classification," *International Conference on Data Mining*, 2016.

[14] T. Liu and D. Tao, "Classification with noisy labels by importance reweighting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.

[15] W. Gao, L. Wang, and Y. Li, "Risk minimization in the presence of label noise," *AAAI*, 2016.

[16] G. Patrini, F. Nielsen, R. Nock, and M. Carinoi, "Loss factorization, weakly supervised learning and label noise robustness," *International Conference on Machine Learning*, 2017.

[17] Y. Wang, J. Sharpnack, and R. Tibshirani, "Trend filtering on graphs," *Journal of Machine Learning Research (JMLR)*, 2016.

[18] Q. Mao, L. Wang, and Y. Sun, "Dimensionality reduction via graph structurelearning," *KDD*, 2015.

[19] Y. Liu and H. Guo, "Peer Loss Functions-Learning from Noisy Labels without Knowing Noise Rates," *International Conference on Machine Learning*, 2020.

[20] Y. Yao, T. Liu, B. Han, M. Gong, J. Deng, G. Niu, and M. Sugiyama, "Dual T: Reducing Estimation Error for Transition Matrix in Label-noise Learning," *NeurIPS*, 2020.

TABLE VI
ABLATION STUDY RESULTS IN TERMS OF TESTING ACCURACY(%) ON CIFAR-10 (C10) AND CIFAR-100 (C100).

Dataset	C10	C10	C10	C10	C10	C100	C100	C100	C100
Noise type	Sym.	Sym.	Sym.	Sym.	Asym.	Sym.	Sym.	Sym.	Sym.
Label Noise, Outliers	(50,10)%	(80,10)%	(50,20)%	(80,20)%	(40,10)%	(50,10)%	(80,10)%	(50,20)%	(80,20)%
URSSL w/o SRGDR	92.8(1.6)	90.7(2.0)	91.5 (0.6)	88.7(0.6)	92.3(0.7)	77.3(0.7)	72.0(0.6)	63.8(0.6)	60.9(0.5)
URSSL w/o robust estimators	93.7(1.6)	92.6(2.0)	93.4(0.4)	92.1(0.8)	91.6(0.5)	77.6(0.7)	74.9(0.3)	61.2(0.6)	61.7(0.5)
URSSL w/o network regularization	93.6(1.6)	92.7(2.0)	93.1(0.4)	92.3(0.5)	93.8(0.5)	79.3(0.7)	76.2(0.7)	64.7(0.6)	63.1(0.6)
URSSL	**95.7(0.6)**	**94.5(0.7)**	**95.3(0.6)**	**94.1(0.5)**	**95.1(0.3)**	**79.6(0.3)**	**77.7(0.4)**	**65.5(0.4)**	**63.7(0.6)**

TABLE VII
COMPARISON OF THE TRAINING AND INFERENCE TIME (HOURS) OF URSSL ON THE CIFAR10 DATASET WITH SEVERAL STATE-OF-THE-ART APPROACHES EVALUATED ON A SINGLE NVIDIA V100 GPU.

Coteach[29]	Co-mine[42]	Coteach+[43]	DivideMix [27]	Decoupling [41]	MentorNet [39]	DPSSL[31]	CAFA	URSSL
4.3h	6.0h	5.2h	5.8h	6.5h	5.3h	8.1h	4.9h	4.7h
0.26h	0.31h	0.29h	0.28h	0.33h	0.35h	0.43h	0.35h	0.27h

[21] Q. Mao, L. Wang, I. W. Tsang, and Y. Sun, "Principal Graph and Structure Learning Based on Reversed Graph Embedding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.

[22] J. A. Lee and M. Verleysen, "Shift-invariant similarities circumvent distance concentration in stochastic neighbor embedding and variants," *International Conference on Computational Science*, 2011.

[23] M. Nikolova and M. Ng, "Analysis of half-quadratic minimization methods for signal and image recovery," *SIAM Journal on Scientific computing*, 2005.

[24] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, pp. 183–202, 2009.

[25] R. He, W. Zheng, and B. Hu, "Maximum correntropy criterion for robust face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011.

[26] Y.-F. Li, J. T. Kwok, and Z.-H. Zhou, "Semi-supervised learning using label mean," *ICML*, 2009.

[27] J. Li, R. Socher, and S. Hoi, "DIVIDEMIX: Learning with noisy labels as semi-supervised learning," *International Confernce on Learning Respresentation*, 2020.

[28] J. Li, Y. Wong, Q. Zhao, and M. Kankanhalli, "Learning to Learn from Noisy Labeled Data," *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[29] X. Yu, B. Han, J. Yao, G. Niu, I. Tsang, and M.Sugiyama, "How does disagreement help generalization against label corruption?" *International Conference on Machine Learning*, 2019.

[30] Z. Huang, C. Xue, B. Han, J. Yang, and C. Gong, "Universal semi-supervised learning," *Neural Information Processing Systems*, 2021.

[31] Y. Xu, J. Ding, L. Zhang, and S. Zhou, "DP-SSL: Towards Robust Semi-supervised Learning with A Few Labeled Samples," *Neural Information Processing Systems*, 2021.

[32] P. Chen, B. Liao, G. Chen, and S. Zhang, "Understanding and utilizing deep neural networks trained with noisy labels," *International Conference on Machine Learning*, 2019.

[33] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," *AAAI*, 2017.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mapping in deep residual network," *ECCV*, 2016.

[35] X. Chen and B. Wujek, "AutoDAL: Distributed Active Learning with Automatic Hyperparameter Selection," *AAAI*, 2020.

[36] W. Ha and R. Barber, "Robust PCA with compressed data," *NIPS*, 2015.

[37] D. Tanaka, D. Ikami, T. Yamasaki, and K. Aizawa, "Joint optimization framework for learning with noisy labels," *International Conference on Computer Vision and Patterson Recognition*, 2018.

[38] Z. Zhang, F. Li, L. Jia, J. Qin, L. Zhang, and S. Yan, "Robust Adaptive Embedded Label Propagation With Weight Learning for Inductive Classification," *IEEE Transactions on Neural Networks and Learning Systems*, 2018.

[39] L. Jiang, Z. Zhou, T. Leung, L. Li, and L. Fei-Fei, "Mentornet: Learning datadriven curriculum for very deep neural networks on corrupted labels," *International Conference on Machine Learning*, 2018.

[40] G. Patrini, A. Rozza, A. Menon, R. Nock, and L. Qu, "Making deep neural networks robust to label noise: A loss correction approach," *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[41] E. Malach and S. Shwartz, "Decoupling "when to update" from " how to update"," *Neural Information Processing Systems*, 2017.

[42] X. Wang, S. Wang, J. Wang, H. Shi, and T. Mei, "Co-Mining: Deep Face Recognition with Noisy Labels," *International Conference on Computer Vision*, 2019.

[43] X.Yu, B. Han, J. Yao, G. Niu, I. W. Tsang, and M. Sugiyama, "How does disagreement help generalization against label corruption?" *International Conference on Machine Learning*, 2019.