Efficient and Robust Semi-supervised Learning Over a Sparse-Regularized Graph

Hang Su[1(✉)], Jun Zhu[1], Zhaozheng Yin[2], Yinpeng Dong[1], and Bo Zhang[1]

[1] State Key Lab of Intelligent Technology and Systems,
Tsinghua National Lab for Information Science and Technology,
Department of Computer Science and Technology,
Center for Bio-Inspired Computing Research, Tsinghua University, Beijing, China
{suhangss,dcszj,dcszb,dongyp13}@tsinghua.edu.cn
[2] Department of Computer Science,
Missouri University of Science and Technology, Rolla, USA
yinz@mst.edu

Abstract. Graph-based Semi-Supervised Learning (GSSL) has limitations in widespread applicability due to its computationally prohibitive large-scale inference, sensitivity to data incompleteness, and incapability on handling time-evolving characteristics in an open set. To address these issues, we propose a novel GSSL based on a batch of informative beacons with sparsity appropriately harnessed, rather than constructing the pairwise affinity graph between the entire original samples. Specifically, (1) beacons are placed automatically by unifying the consistence of both data features and labels, which subsequentially act as indicators during the inference; (2) leveraging the information carried by beacons, the sample labels are interpreted as the weighted combination of a subset of characteristics-specified beacons; (3) if unfamiliar samples are encountered in an open set, we seek to expand the beacon set incrementally and update their parameters by incorporating additional human interventions if necessary. Experimental results on real datasets validate that our algorithm is effective and efficient to implement scalable inference, robust to sample corruptions, and capable to boost the performance incrementally in an open set by updating the beacon-related parameters.

Keywords: Semi-supervised learning · Beacon · Sparse representation · Online learning

1 Introduction

In the era of information deluge, Semi-Supervised Learning (SSL) [1,2], which implements inference by combining a limited amount of labeled data and abundant unlabeled data in open sources, is a promising direction to cope with the

This research is partly supported by the National Basic Research Program (973 Program) of China (Nos. 2013CB329403, 2012CB316301), National Natural Science Foundation of China (Nos. 61571261, 61322308, 61332007), China Postdoctoral Science Foundation (No. 2015M580099). Zhaozheng Yin is supported by NSF CAREER Award IIS-1351049 and NSF EPSCoR grant IIA-1355406.

B. Leibe et al. (Eds.): ECCV 2016, Part VIII, LNCS 9912, pp. 583–598, 2016.
DOI: 10.1007/978-3-319-46484-8_35

flood of big data. Among various SSL methods, Graph-based Semi-Supervised Learning (GSSL) [3,4] is an appealing paradigm thanks to the prevalence of graph data and its good capability in exploiting intrinsic manifold structures.

Recent years have witnessed significant advances in GSSL, including Mincut [5], Random Walk [6,7], Manifold Regularization [8], Gaussian Fields and Harmonic Functions (GFHF) [9], and Learning with Local and Global Consistency (LLGC) [10]. Nevertheless, the algorithms are often sensitive to data noise and improper parameter settings [11,12], i.e., the graph structures may be changed dramatically due to the corruption of features or shift of global hyper-parameters. To address these issues, Cheng et al. [11] proposed an ℓ_1-graph, which is robust to data noise and adaptive to graph structures. However, these algorithms are actually designed for small or medium sized data; the high computational complexity blocks their widespread applicability to real-life problems.

To temper the time complexity, a lot of efforts have been made during the past years, e.g., Nystrom approximation [13], the eigenfunction approximation [14], ensemble projection [15], etc. Among these works, anchor-based algorithms are attractive [16,17], which construct a tractable large graph by coupling anchor-based label prediction and adjacency matrix design. However, anchors in these methods are obtained in two separate steps—anchors are placed in the feature domain only based on the feature information but neglecting the useful knowledge in labels; the anchor labels are then estimated by propagating the labels of human-annotated samples whose locations in the feature domain are already fixed. We would expect that these two steps can mutually enhance each other if they are properly unified and learned jointly.

Above all, the aforementioned algorithms assume that queries are drawn from a closed pool and the properties of training and testing samples are the same. Unfortunately, this assumption may not be valid in many real-world scenarios, where the training and testing data may be collected under different experimental conditions and therefore often exhibit differences in their statistics; and properties of samples may gradually change over time thus incomplete knowledge is present at the training phase. In this case, a classifier learned from the initial labeling tends to result in more and more misclassifications if no further knowledge is provided or no update paradigm is applied.

1.1 Our Proposal

To address the above issues (i.e., *data noise, time complexity and statistics shift*), we propose an ℓ_1-Beacon Graph based semi-supervised algorithm, which places a batch of characteristic-specific beacons in the feature domain, and represent the original samples with a subset of beacons. Prediction on missing labels can be implemented with label fusion of the corresponding beacons.

To mitigate the computational bottleneck, the label prediction is implemented by weighted averaging the soft labels of a subset of *beacons*, which is a concept in large-scale network analysis [18]. In this paper, we use *beacons* to represent the super-nodes whose characteristics are propagated from the human specified information, and ultimately facilitate the sample inference. Specifically, the beacons are generated automatically by minimizing the sample-to-beacon reconstruction error while preserving the label consistence jointly. The resultant beacons therefore behave as indicators to guide the inference procedure, i.e., the information provided by human annotations is propagated to the beacons and "lights" their indications or soft labels. Different from the orthogonal anchor planes in [19], our method does not requires orthogonal planes to represent/code samples.

In the testing phase with voluminous or streaming data to handle, the label inference is implemented by setting up a relational connection between the original samples and the most relevant beacons, which are identified by enforcing the sparsity. The ℓ_1-*regularization* also offers robustness to the corrupted or incomplete sample features [11], which is an inevitable nightmare [20] for the large-scale data analysis. The main reason is because the sample-to-beacon relationship can be estimated appropriately by making use of the abundant information embedded in the uncorrupted feature entries under the sparsity constraint.

When unfamiliar samples are encountered in the *open-set* inference, e.g., the reconstruction error is above a user-specified threshold, we seek to expand the beacon set and update their characteristic parameters dynamically by incorporating additional human interventions. Consequently, the performance is boosted incrementally with a small amount of computation for the unseen data with time-evolving statistics.

Compared with the anchor-based algorithms in [16,17], the proposed algorithm has the major novelties below

- To address the beacon construction, we propose to learn the beacons by utilizing both the label information and feature information jointly, which generalizes the K-means clustering anchors in [16,17] and provides a more flexible representation for data lying in a complex manifold.
- To explore the sample-to-beacon relationship, we derive the neighboring beacons of a sample and the corresponding relationship weights automatically by solving an ℓ_1-norm regularized problem, which yields an adaptive and flexible representation especially for data in a complex high-dimensional manifold. In contrast, the samples are represented with s neighboring anchors in [16], which may incur significant performance degeneration if the global parameter "s" is set improperly. With sparsity properly harnessed, our method also offers adaptations of local neighborhood structures and robustness for the corrupted sample features.
- To address the issues in the open-set inference, we propose to expand the beacon set and update their characteristics incrementally, which offers advantages to handle the mismatch between training and testing data.

In summary, the proposed algorithm is much more robust to data noise, provides a more adaptive and stable graph construction to local neighborhood structures, needs fewer beacons to realize the comparable performance, and boosts the performance incrementally when unfamiliar samples appear.

2 Construction of ℓ_1-Beacon Graph

Semi-supervised learning typically involves a dataset that consists of N_l labeled data $\mathcal{L} \triangleq \{(\mathbf{x}_l, \mathbf{y}_l)\}_{l=1}^{N_l}$ and N_u unlabeled data $\mathcal{U} \triangleq \{\mathbf{x}_u\}_{u=1}^{N_u}$, where $N_u \gg N_l$ and $N = N_l + N_u$. Label propagation algorithms [9,10] entangle all these samples, and build a huge graph to model the pairwise similarity between samples in the entire dataset, which require to calculate the inverse of a large Laplacian matrix with the cubic time complexity $O(N^3)$. Therefore, it becomes an unbearable computational burden for processing gigantic even medium-sized datasets.

In this case, it is desirable to develop more efficient algorithms by taking advantage of both the labeled and a portion of the unlabeled data to build a training dataset $\mathcal{X}^{\text{train}}$, and train a classification model to handle unseen data outside the training data set. Much fewer training samples are used for these models and thus more efficient for the label prediction.

In this paper, we propose to generate a batch of "beacons", which behave as indicators to guide the inference. Original samples are represented by a linear combination of the beacons, resulting in a sample-to-beacon relationship matrix. The predicted labels of samples are inferred as the weighted combination of a subset of beacons as,

$$\mathbf{Y} = \mathbf{F}\mathbf{Z}, \text{ with } \mathbf{Z} \in \mathbb{R}^{M \times N},\ M \ll N, \tag{1}$$

where $\mathbf{Y}$ is the prediction label matrix with each column being the label of a specific sample; $\mathbf{B} = [\mathbf{b}_1, \cdots, \mathbf{b}_M]$ is a beacon set and $\mathbf{F} = [f(\mathbf{b}_1), \cdots, f(\mathbf{b}_M)]$ is the label matrix with each column corresponding to the label of a beacon; and $\mathbf{Z} = [\mathbf{z}_1, \cdots, \mathbf{z}_N]$ is the weight matrix in which each column indicates the sample-to-beacon relationship for a specific sample. To solve the problem in Eq. (1), we need to

- Determine the informative beacons $\mathbf{B}$ along with their labelling characteristics $\mathbf{F}$ effectively; and
- Calculate the sample-to-beacon relationship matrix $\mathbf{Z}$ efficiently despite the incompleteness and corruption existing in data.

2.1 Design of Informative Beacons

In [16,17], the authors proposed to generate K anchors using the clustered centers by K-means, and estimate the corresponding relationship matrix by representing samples as linear combinations of s nearest neighboring anchors. However, it is difficult to determine the optimal parameters of K and s in advance. For instance, the distribution of samples and their neighboring anchors may vary at different areas in the feature domain, which results in distinctive neighborhood structures for each sample. In this case, the graph generated via local anchor embedding [16] may introduce unreasonable neighborhood structures due to the improper parameters. In many cases, these unfavorable structures incur a significant performance degeneration since the labels may be propagated via those edges across samples belonging to different classes.

To address these issues, we seek a beacon set that yields a flexible and adaptive representation by utilizing the ℓ_1-norm regularization. Additionally, the beacon generation and their corresponding characteristic estimations are unified within a framework with both the features and label information harnessed, thereby encouraging their mutual enhancements.

With a unified representation, we denote the beacon-related parameters as $\varPsi = [\mathbf{B}; \mathbf{F}]$ with each column corresponding to a specific beacon embedding its related characteristic information. Therefore, $\mathbf{B} = \mathbf{S}_b \varPsi$ with $\mathbf{S}_b = [\mathbf{I}_d, \mathbf{0}] \in \mathbb{R}^{d\times(d+c)}$, and $\mathbf{F} = \mathbf{S}_f \varPsi$ with $\mathbf{S}_f = [\mathbf{0}, \mathbf{I}_c] \in \mathbb{R}^{c\times(d+c)}$. d and c are the sample dimension and corresponding numbers of classes, respectively; and $\mathbf{I}_d$ and $\mathbf{I}_c$ are the identity matrices with proper sizes.

By taking both the feature and label information into account, the beacon generation can be derived by minimizing the risk functions for both labeled and unlabeled data and also preserving the global graph smoothness as

$$\begin{aligned}(\varPsi^*, \mathbf{Z}^*) = \arg\min_{\varPsi,\mathbf{Z}} R_{\mathcal{L}}(\varPsi, \mathbf{Z}) + R_{\mathcal{U}}(\varPsi, \mathbf{Z}) + \lambda R_{\mathbf{Y}}(\varPsi, \mathbf{Z}),\\ \text{s.t. } \forall i \in [1, N],\ \|\mathbf{z}_i\|_1 \leq T,\ \mathbf{z}_i \geq \mathbf{0},\ \mathbf{y}_i = \mathbf{S}_f \varPsi \mathbf{z}_i.\end{aligned} \tag{2}$$

Specifically, the risk function on the labeled set is defined as

$$R_{\mathcal{L}}(\varPsi, \mathbf{Z}) = \sum_{i=1}^{N_l} \left\| \begin{bmatrix} \mathbf{x}_i \\ \mathbf{y}_i \end{bmatrix} - \varPsi \mathbf{z}_i \right\|_2^2, \tag{3}$$

which jointly penalizes the reconstruction error in the feature domain and preserves the consistence in the label domain; the risk function on the unlabeled set is defined as

$$R_{\mathcal{U}}(\varPsi, \mathbf{Z}) = \sum_{i=1}^{N_u} \|\mathbf{x}_i - \mathbf{S}_b \varPsi \mathbf{z}_i\|_2^2, \tag{4}$$

which is the residual error on all unlabeled samples; and the graph smoothness regularization is defined as

$$R_{\mathbf{Y}}(\varPsi, \mathbf{Z}) = \sum_{i,j}^{N} \|\mathbf{y}_i - \mathbf{y}_j\|_2^2 w_{ij}, \tag{5}$$

where $\mathbf{y}_i$ and $\mathbf{y}_j$ are encouraged to be similar if $\mathbf{x}_i$ and $\mathbf{x}_j$ are close in the intrinsic geometry of the feature domain.

In Eq. (2), $\mathbf{Z} = [\mathbf{z}_1, \cdots, \mathbf{z}_N]$ is the weight matrix with each column corresponding to the sample-to-beacon relationship of a specific sample; T is the sparsity level, which is related to the number of beacons that are chosen for the representation; $w_{ij} = \mathbf{z}_i^T \mathbf{z}_j$ is the pairwise affinity between $\mathbf{x}_i$ and $\mathbf{x}_j$, which is measured in terms of correlation (inner product).

When deriving the problem in Eq. (2), the beacons are placed in the feature domain and lightened up simultaneously by providing its label characteristics.

By incorporating the feature and label information together, the generated beacons are therefore consistent with both labels and features. Additionally, the results are also benefited from the graph smoothness, which favors the label consistence when samples share similar features.

By introducing the ℓ_1-norm regularization on the sample-to-beacon relationship $\mathbf{z}_i$, the most relevant beacons are selected automatically to represent the samples. Therefore, it provides a more adaptive representation for data lying in a complex manifold by reducing the spurious connections between the dissimilar sample-to-beacon connections. The nonnegative property on $\mathbf{z}_i$ further guarantees a positive semi-definite Laplacian matrix when inferring the sample-to-sample affinity via the sample-to-beacon mapping, which is of importance to ensure a global optimum of Graph-based Semi-Supervised Learning [1].

2.2 Optimization Algorithm

The problem in Eq. (2) is convex with respect to each of the two variables Ψ and $\mathbf{Z}$ when the other one is fixed. It can be solved by alternately minimizing one variable while keeping the other one fixed.

We construct an affinity matrix $\mathbf{W} = [w_{ij}] = \mathbf{Z}^T\mathbf{Z}$ which characterizes the pairwise similarity between samples in the training set, resulting in a similarity graph $\mathcal{G}$ with the affinity matrix $\mathbf{W}$. In this case, the corresponding Laplacian matrix is $\mathbf{L} = \mathbf{D} - \mathbf{W}$ with $\mathbf{D}$ being the diagonal degree matrix as $\mathbf{D}(j,j) = \sum\limits_{i=1}^{N} \mathbf{W}(i,j)$. Therefore, the graph smoothness term in Eq. (5) can be rewritten as

$$R_{\mathbf{Y}}(\Psi, \mathbf{Z}) = \sum_{i=1}^{N} \mathbf{y}_i^T \mathbf{L} \mathbf{y}_i = \sum_{i=1}^{N} \mathbf{z}_i^T \mathbf{F}^T \mathbf{L} \mathbf{F} \mathbf{z}_i \tag{6}$$

$$= \mathrm{tr}(\mathbf{Y}\mathbf{L}\mathbf{Y}^T) = \mathrm{tr}(\mathbf{F}\mathbf{Z}\mathbf{L}\mathbf{Z}^T\mathbf{F}^T). \tag{7}$$

Using Eqs. (6) and (7), the problem in Eq. (2) can be solved by alternately solving the following two problems as

P1: Solving $\mathbf{Z}$ by fixing Ψ:

$$\mathbf{Z}^* = \arg\min_{\mathbf{Z}=[\mathbf{z}_i]} \left\{ \sum_{i=1}^{N_l} \| [\mathbf{x}_i; \mathbf{y}_i] - \Psi \mathbf{z}_i \|_2^2 + \sum_{i=1}^{N_u} \| \mathbf{x}_i - \mathbf{B}\mathbf{z}_i \|_2^2 + \lambda \sum_{i=1}^{N} \mathbf{z}_i^T \mathbf{F}^T \mathbf{L} \mathbf{F} \mathbf{z}_i \right\},$$
$$\text{s.t. } \forall i, \ \|\mathbf{z}_i\|_1 \le T, \ \mathbf{z}_i \ge \mathbf{0}, \tag{8}$$

where $\mathbf{B}$ and $\mathbf{F}$ are the sub-matrices of Ψ corresponding to the feature and label domains, respectively; and

P2: Solving Ψ by fixing $\mathbf{Z}$:

$$\Psi^* = \arg\min_{\Psi} \left\{ \| [\mathbf{X}_l; \mathbf{Y}_l] - \Psi \mathbf{Z}_l \|_2^2 + \| \mathbf{X}_u - \mathbf{B}\mathbf{Z}_u \|_2^2 + \lambda \mathrm{tr}(\mathbf{F}\mathbf{Z}\mathbf{L}\mathbf{Z}^T\mathbf{F}^T) \right\}$$
$$\text{s.t. } \mathbf{B} = \mathbf{S}_b \Psi, \ \mathbf{F} = \mathbf{S}_f \Psi, \tag{9}$$

where $\mathbf{X}_l = [\mathbf{x}_1, \cdots, \mathbf{x}_{N_l}]$ and $\mathbf{X}_u = [\mathbf{x}_1, \cdots, \mathbf{x}_{N_u}]$ are corresponding to the labeled and unlabeled samples, respectively; $\mathbf{Y}_l = [\mathbf{y}_1, \cdots, \mathbf{y}_{N_l}]$ is the indicator matrix by stacking sample labels in column; $\mathbf{Z}_l$ and $\mathbf{Z}_u$ are corresponding to the sub-matrix of $\mathbf{Z}$ related to the labeled and unlabeled subsets, respectively.

In order to solve $P1$, we propose to calculate $\mathbf{z}_i$ iteratively using the efficient interior-point method [21], which jointly preserves consistence of both the feature and label information. Afterwards, the Laplacian matrix $\mathbf{L}$ in Eq. (9) is updated with the corresponding sample-to-beacon relationship matrix $\mathbf{Z}$. Moreover, the objective function in $P2$ can be rewritten as

$$(\mathbf{B}^*, \mathbf{F}^*) = \arg\min_{\mathbf{B},\mathbf{F}} g(\mathbf{B}, \mathbf{F}) = \arg\min_{\mathbf{B},\mathbf{F}} \|\mathbf{X} - \mathbf{B}\mathbf{Z}\|_2^2 + \|\mathbf{Y}_l - \mathbf{F}\mathbf{Z}_l\|_2^2 + \lambda \mathrm{tr}(\mathbf{F}\mathbf{Z}\mathbf{L}\mathbf{Z}^T\mathbf{F}^T). \tag{10}$$

Applying the cyclic property of trace and differentiating Eq. (10) with respect to $\mathbf{B}$ and $\mathbf{F}$, the partial derivatives of Eq. (10) is

$$\frac{\partial g}{\partial \mathbf{B}} = -2(\mathbf{X} - \mathbf{B}\mathbf{Z})\mathbf{Z}^T, \ \frac{\partial g}{\partial \mathbf{F}} = -2(\mathbf{Y}_l - \mathbf{F}\mathbf{Z}_l)\mathbf{Z}_l^T + 2\lambda \mathbf{F}\mathbf{Z}\mathbf{L}\mathbf{Z}^T. \tag{11}$$

Setting the derivatives in Eq. (11) to zeros yields the optimal solution as

$$\mathbf{B}^* = \mathbf{X}\mathbf{Z}^T(\mathbf{Z}\mathbf{Z}^T)^{-1}, \ \mathbf{F}^* = \mathbf{Y}_l\mathbf{Z}_l^T(\mathbf{Z}_l\mathbf{Z}_l^T + \lambda\mathbf{Z}\mathbf{L}\mathbf{Z}^T)^{-1}. \tag{12}$$

Hereby, the optimum of Ψ in $P2$ is obtained as $\Psi^* = [\mathbf{B}^*; \mathbf{F}^*]$. We alternately optimize the $P1$ in Eq. (8) and $P2$ in Eq. (9) until convergence. It is noted that, in Eq. (12), the inversion is computed on a rather small matrix sized $M \times M$ efficiently, rather than a huge matrix sized $N \times N$ in the previous GSSLs [9,10].

3 Inductive Inference

After obtaining the M beacons $\mathbf{B}$ along with their characteristics $\mathbf{F}$, the scalable inference in the testing data is implemented by label fusion of those beacons that are linked to the test data. For any testing sample $\tilde{\mathbf{x}}_i$, we propose to determine its neighborhood structure along with the strength of sample-to-beacon association by solving the following optimization problem with the sparsity constraint as

$$\tilde{\mathbf{z}}^* = \arg\min_{\tilde{\mathbf{z}}_i} \|\tilde{\mathbf{x}}_i - \mathbf{B}\tilde{\mathbf{z}}_i\|_2^2, \ \text{s.t.} \ \|\tilde{\mathbf{z}}_i\|_1 \leq T, \ \tilde{\mathbf{z}}_i \geq \mathbf{0}, \tag{13}$$

where $\tilde{\mathbf{z}}_i$ denotes the relationship between sample i and beacons. After solving $\tilde{\mathbf{z}}_i$ iteratively using the efficient interior-point method [21], the identity of sample i is obtained via label fusion by plugging $\mathbf{F}$ and $\tilde{\mathbf{z}}_i$ into Eq. (1) as

$$\tilde{\mathbf{y}}_i^* = \mathbf{F}\tilde{\mathbf{z}}_i^*. \tag{14}$$

The hard label vector can be obtained simply by converting the maximum value in each $\mathbf{y}_i^*$ into 1 and the others into 0.

With sparsity appropriately harnessed in Eq. (13), the most relevant beacons are chosen to describe the samples, which improves the performance in terms of efficiency, accuracy and robustness to noise. By introducing the beacons to the feature domain, it is not necessary to estimate the pairwise sample affinity in the sheer volume of testing data [11,12], which reduces the time cost significantly. Additionally, the beacon-based inference can be conducted via distributed computing by sharing beacon information across different servers, which is of practical value in large-scale data analysis.

4 Incremental Update of Beacons in Open-Set

To address the characteristics-evolving issue in the open-set inference, it is worth to consider how to further update the model to boost the performance with a small amount of incremental computation with new data. Specifically, when the model encounters a set of "unfamiliar" samples with new, novel or unknown characteristics that cannot be reconstructed using the existing beacons well (e.g., the reconstruction error is above a threshold as $r(\mathbf{x}_i) = \|\mathbf{x}_i - \mathbf{B}\mathbf{z}_i\|_2^2 > th$), we propose to expand the beacon set by incrementally adding k new beacons $\mathbf{B}_k = [\mathbf{b}_k]$ as $\bar{\mathbf{B}} \triangleq [\mathbf{B}, \mathbf{B}_k]$.

Since the beacon set $\mathbf{B}$ is obtained by solving the problem in Eq. (2), the intuitive way to conduct the beacon update would be to learn the beacons from scratch using the set of identified unfamiliar data $\bar{\mathcal{X}}$ along with the training samples, i.e., substituting $\mathcal{X}^{\text{train}}$ with $\mathcal{X}^{\text{train}} \cup \bar{\mathcal{X}}$ in Eq. (2). Nevertheless, it is inefficient to re-build the model from scratch and it would be nice if the model could be updated incrementally.

In this case, we propose to improve the on-hand beacon set $\mathbf{B}$ incrementally by feeding the unfamiliar samples to handle the time-evolving characteristics. Specifically, we initialize the newly-created k beacons as $\mathbf{B}_k = [\epsilon \mathbf{I}_{d\times k}]$ with ϵ being a positive number that is close to zero; afterwards, the $\mathbf{B}_k$ is updated gradually with the stochastic gradient descent (SGD) algorithm [22] until convergence. We derive the incremental updating algorithm below.

Similarly to Eq. (15), the partial derivatives with respect to the beacon parameter $\bar{\mathbf{B}}$ and its corresponding characteristic parameter $\bar{\mathbf{F}}$ in the unfamiliar sample set are

$$\frac{\partial g}{\partial \bar{\mathbf{B}}} = -2(\bar{\mathbf{X}} - \bar{\mathbf{B}}\bar{\mathbf{Z}})\bar{\mathbf{Z}}^T, \; \frac{\partial g}{\partial \bar{\mathbf{F}}} = -2(\mathbf{Y}_l - \bar{\mathbf{F}}\mathbf{Z}_l)\mathbf{Z}_l^T + \lambda \bar{\mathbf{F}}\bar{\mathbf{Z}}\bar{\mathbf{L}}\bar{\mathbf{Z}}^T, \tag{15}$$

where $\bar{\mathbf{X}} = [\bar{\mathbf{x}}_i]$ represents the samples that are detected to be unfamiliar to the initial beacon set; and $\bar{\mathbf{Z}} = [\bar{\mathbf{z}}_i]$ is the sample-to-beacon relationship matrix with each column corresponding to a specific sample; $\mathbf{Z}_l$ is the sub-matrix of $\bar{\mathbf{Z}}$ related to the labeled subset, which is updated when more human annotations are provided by users on the unfamiliar samples.

In order to update the parameters iteratively, we denote $\bar{\mathbf{X}}_t$ as the novel samples drawn at iteration t, and the beacon set can be updated using the Stochastic Gradient Descent (SGD) as

$$\bar{\mathbf{B}}_t = \bar{\mathbf{B}}_{t-1} - \delta \frac{\partial g}{\partial \bar{\mathbf{B}}_{t-1}} = \bar{\mathbf{B}}_{t-1} + 2\delta(\bar{\mathbf{X}}_t \bar{\mathbf{Z}}_t^T - \bar{\mathbf{B}}_{t-1} \bar{\mathbf{Z}}_t \bar{\mathbf{Z}}_t^T),$$

$$\bar{\mathbf{F}}_t = \bar{\mathbf{F}}_{t-1} - \delta \frac{\partial g}{\partial \bar{\mathbf{F}}_{t-1}} = \bar{\mathbf{F}}_{t-1} + \delta \left(2(\mathbf{Y}_l - \bar{\mathbf{F}}_{t-1}\mathbf{Z}_l)\mathbf{Z}_l^T - \lambda \bar{\mathbf{F}}_{t-1} \bar{\mathbf{Z}}_t \mathbf{L}_t \bar{\mathbf{Z}}_t^T\right), \quad (16)$$

where $\bar{\mathbf{B}}_t$ is the update of the beacon set $\bar{\mathbf{B}}_{t-1}$ at the t_{th} iteration; $\bar{\mathbf{Z}}_t$ is the sample-to-beacon relationship for the labeled samples in the t_{th} iteration; and δ is the learning rate. When the algorithm converges, the optimal beacon-related parameter $\bar{\Psi}$ is obtained as $\bar{\Psi} = [\bar{\mathbf{B}}; \bar{\mathbf{F}}]$. In this paper, we use the original beacon-related parameters $\mathbf{B}$ and $\mathbf{F}$ in Eq. (12) as a warm start.

5 Experiments

In this section, we evaluate the proposed ℓ_1-Beacon Graph based Semi-Supervised Learning algorithm against alternative algorithms in terms of accuracy, time complexity, robustness to data corruption and data incompleteness, and the performance in the open-set inference.

5.1 Datasets

To verify the effectiveness of our algorithm on graph construction and scalable inference, we implement image classification and image segmentation with three real-world benchmark datasets in our experiments. To evaluate our method, we conduct image classification on **MNIST**[1] and **CIFAR**[2], and image segmentation on **CELL**[3].

5.2 Sample Results

Figure 1 shows some examples of image classification. In each experiment, we use intensity of images as visual features, and annotate a small portion of samples in each dataset (1 % for MNIST and 5 % for CIFAR) as seeds for subsequential estimation of beacon characteristics and inference on categories of unlabeled samples. The images with green and red boundaries in Fig. 1 denote the true and false recognitions, respectively. As is observed, most of the images are classified into confident categories except for samples with odd morphological features. False classifications for CIFAR occur when the dominant object does not occupy significant areas in the image.

[1] **MNIST** consists of 70,000 handwritten digits sized 28×28 with 60,000 training ones, http://yann.lecun.com/exdb/mnist/.

[2] **CIFAR** consists of 60,000 32×32 color images in 10 classes, with 6000 images per class, http://www.cs.toronto.edu/~kriz/cifar.html.

[3] **CELL** contains different types of muscle stem cells of a progeroid mouse in time-lapse microscopy sequences, in which each frame contains 50∼800 cells, http://www.celltracking.ri.cmu.edu/downloads.html.

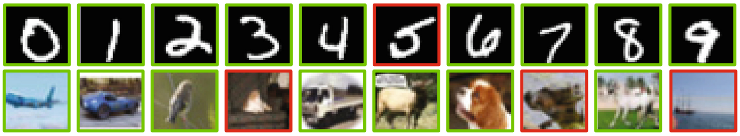


Fig. 1. Sample results of image classification, in which the image with green and red boundaries indicates the true and false recognition, respectively. (Color figure online)

Figure 2 shows some sample results for cell segmentation. For each sequence, each image is first partitioned into superpixels [23]. Cell segmentation is realized by classifying the superpixels into specific classes based on a small portion of annotated superpixels (around 1.5 % in our experiments). As is demonstrated in the results, superpixels corresponding to different cells with different visual characteristics are classified into specific categories, resulting in a cell segmentation with high qualities.

5.3 Comparison Methods

In order to evaluate the proposed algorithm, we compare our ℓ_1-*Beacon graph* based algorithm against alternative learning algorithms with respect to beacon-based and sample-based methods.

- **Beacon**-based algorithms. We generate beacons $\mathbf{B}$ using the centers of K-means clustering, and then calculate the sample-to-beacon relationship matrix by the Local Anchor Embedding (LAE) (K-means LAE) [16], and Nadaraya-Watson Kernel regression (K-means Kernel) [17], respectively.
- **Sample**-based algorithms. Besides the beacon-based algorithms, we also implement the classification based on the sample-based algorithms. Specifically, we construct the *KNN graph* [1] and the ϵ-*graph* [2], based on which the classification is implemented with the label propagation algorithm [9]. Additionally, the label propagation is also conducted on the ℓ_1-*graph* in [11].

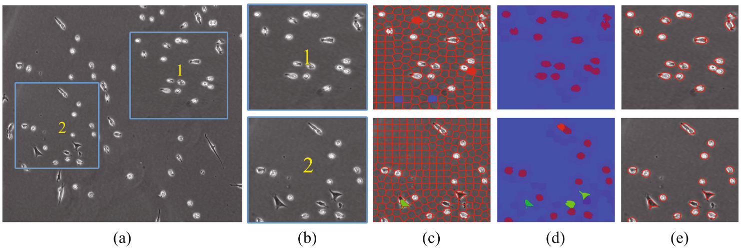


Fig. 2. Sample results of image segmentation. (a) Input phase contrast microscopy images; (b) Zoom-in sub-images; (c) Sample selection and annotation over the superpixels; (d) Soft classification results based on label propagation with human annotations. (e) Cell segmentation by finding the labels with the maximum likelihood, and grouping the neighboring superpixels with the same labels.

To reduce the bias in evaluation, the results are averaged over 10 trials on the testing dataset based on different subsets of seed labels.

5.4 Quantitative Evaluation

Classification Accuracy. In this section, we evaluate the performance against alternative methods after setting the percentage of beacons over samples as 5 % for MNIST, 30 % for CIFAR and 2.5 % for CELL. The optimal essential parameters for each algorithm (e.g., T^* for ℓ_1-Beacon, k_n^* for K-means LAE, ϵ^* for ϵ-graph) are obtained via grid-search. The performance in terms of classification accuracy is reported in Table 1, which demonstrates that all the approaches are comparable in each dataset with the optimal parameters. In some cases, the ϵ-Graph and KNN graph methods outperform other methods mainly because they perform transductive inference and predict the labels by taking all the unlabeled samples into account; however, they lack the capability to handle the samples outside the training datasets, and result in an unbearable burden for large-scale applications.

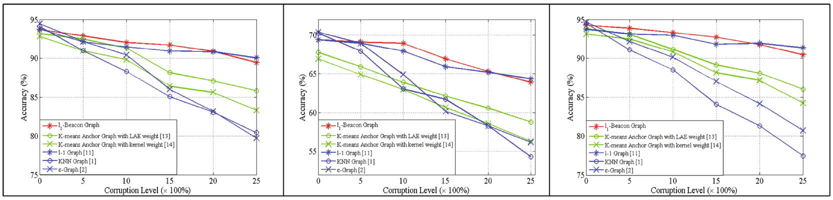
Table 1. Quantitative comparison in terms of accuracy (%)

	Our ℓ_1-Beacon	K-means LAE	K-means Kernel	ℓ_1-Graph	KNN Graph	ϵ-Graph
MNIST	94.87	91.86	92.12	92.78	93.99	95.06
CIFAR	72.53	65.72	65.34	70.21	70.61	73.04
CELL	95.13	92.33	92.87	94.33	95.46	94.90

Robustness to Improper Parameter. In this section, we demonstrate that the proposed algorithm is more robust to the sub-optimal parameters. In each experiment, we sample a batch of sub-optimal parameters by deviating from the optimal ones (T^* for ℓ_1-Beacon, k_n^* for K-means LAE, ϵ^* for ϵ-graph) with up-to 50 % offset, i.e., the sub-optimal parameters $\phi^- = (\epsilon^-, k^-, T^-)$ with four alternative options as $\phi^- = (0.5, 0.75, 1.25, 1.5)\phi^*$, and repeat the experiments with the sub-optimal parameters. The results of mean and standard deviation based on the sub-optimal parameters are reported in Table 2. Compared to the optimal results in Table 1, performance degeneration for the ℓ_1-Beacon graph is not obvious and the deviation is small when the critical parameters are not set optimally. The main reason is because our algorithm still searches for the most informative beacons in the training dataset regarding to the suboptimal parameter setting and links individual samples to their most relevant beacons by a sample-to-beacon relationship matrix. However, the performance degenerates significantly with a larger deviation for alternative methods since noisy information is involved and graph structures are changed due to improper parameters. The property of parameter robustness offers advantages to practical applications, since the parameter sensitiveness is an essential issue for graph-based semi-supervised learning algorithms.

Table 2. Comparison of accuracy with sub-optimal parameters (%)

	Our ℓ_1-Beacon	K-means LAE	K-means Kernel	ℓ_1-Graph	KNN Graph	ϵ-Graph
MNIST	93.62 ± 1.17	89.17 ± 2.23	88.16 ± 3.77	89.83 ± 1.38	84.45 ± 6.75	$85.46\pm$ 8.09
CIFAR	70.86 ± 2.01	61.65 ± 2.89	60.11 ± 5.66	67.03 ± 2.93	65.26 ± 8.66	63.23 ± 9.75
CELL	92.99 ± 2.14	$87.01\pm$ 4.02	84.96 ± 4.98	$89.76\pm$ 2.73	$85.07\pm$ 7.75	$84.22\pm$ 9.33



(a) MNIST (b) CIFAR (c) CELL

Fig. 3. Performance comparison on corrupted data. (Color figure online)

Robustness to Corrupted Samples. To test the robustness of algorithms regarding to sample corruption, we randomly corrupt a portion of entries of the feature vector in the testing samples (i.e., replace their values with random values drawn from a uniform distribution).

As shown in Fig. 3, the performance of our algorithm degrades around 5 %, when 25 % entries of feature vectors are corrupted (red curves). A reasonable explanation is that noise corrupts only a fraction of the feature vector and is therefore sparse in the standard beacons. In this case, the information provided by the uncorrupted entries still offers a good opportunity to estimate the relationship between the samples and beacons. Due to the same reason, ℓ_1-graph [11] is also robust to the noisy features (blue curves with star markers). As a comparison, if ℓ_2 minimization is used to represent corrupted samples, most of the sample-to-beacon relationship matrix may be corrupted [11,17], which will lead to a significant performance degeneration (green curves). Moreover, the performance for the transductive inference based on GFHF [1,2] (blue curves with circle and cross markers) undergoes a significant performance degeneration since the noisy samples introduce too much misleading information and the graph structure is changed greatly if no error suppression paradigm is involved.

Time Complexity. We summarize the time complexity of all methods in Table 3. The label propagation based on KNN and ϵ-graph is of high computational cost due to the matrix inversion operation with complexity $O(N^3)$, where N is the sample number. Our proposed ℓ_1-Beacon is comparable to the K-means anchor-based methods with time complexity $O(M^2N)$ [16], since the ℓ_1 optimization can be implemented efficiently with an empirically complexity $O(M^2N^{1.3})$ [21], where M is the number of beacons ($M \ll N$). However, three sample-based methods are infeasible for larger dataset, e.g., MNIST, since the time cost is rather expensive. For example, in ℓ_1-graph, if pairwise ℓ_1

Table 3. Comparison of time complexity (second)

	ℓ_1-Beacon	K-means LAE	K-means Kernel	ℓ_1-Graph	KNN Graph	ϵ-Graph
MNIST	1103.27	607.32	652.55	4367.90	3616.35	3435.23
CIFAR	2480.35	1932.67	1733.62	8237	9970.52	10322.83
CELL	54.10	48.35	41.36	138.22	1324.71	1237.64

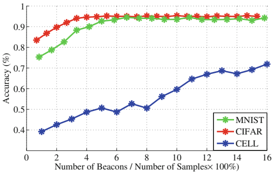


Fig. 4. Accuracy vs. Number of updated beacons

Table 4. Comparison of accuracy

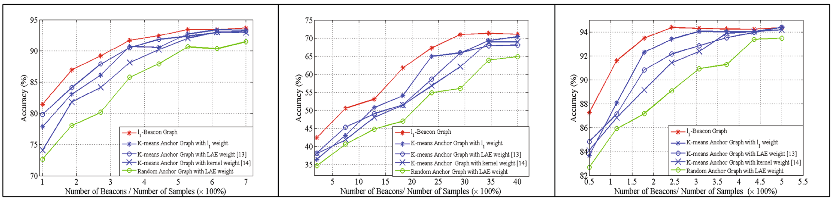
	MNIST	CIFAR	CELL
ℓ_1-Beacon Update	93.14	72.03	95.04
ℓ_1-Beacon	82.50	39.06	76.13
K-means LAE [16]	80.17	36.45	72.06
K-means Kernel [17]	78.31	37.13	73.93
ℓ_1-Graph [11]	84.01	39.61	75.03
KNN [1]	84.99	39.13	77.31
ϵ-Graph [2]	83.30	38.29	78.60

optimization between all samples is implemented, resulting in $O(N^{3.3})$ complexity by particularly setting $M = N$ for ℓ_1-Beacon Graph.

Classification in Open Set. In order to validate the performance of the proposed algorithm in open set, we use only 10 % of the samples in total for the initial beacon training. Therefore, it is expected that there exist some "unfamiliar" samples with high probability due to the incomplete coverage of the feature space, which shares similar properties with the open-set inference. The results are shown in Fig. 4. As is expected and observed, the accuracies are improved as the beacon set expanding and updating, since more informative beacons are involved to handle the samples whose statistics are not present during the initial training phrase.

The comparison in terms of the converged optimal accuracy is reported in Table 4, which demonstrates that the performance benefits substantially from the updating of beacon set. The main reason is that the initial beacons learnt from the initial samples cannot cover the entire feature space, and may lead to improper sample-beacon couplings during the inference. It is also noted that the proposed algorithm allows to load only a small portion of data, and implement the inference incrementally by updating the characteristics of beacons dynamically as data arrive continuously. Therefore, it is useful when analyzing an enormous volume of data in a limited memory.

Performance Versus Beacon Number. Finally, we study the performance versus the number of beacons (M) which is the most critical parameter for the beacon-based algorithms. Figure 5 reveals that the performance is significantly



(a) MNIST (b) CIFAR (c) CELL

Fig. 5. The classification accuracy versus (beacon number/sample number). (Color figure online)

improved as the beacon number increases. However, much fewer beacons are needed to realize a comparable performance for our algorithm (red curves) compared to the K-means centers (blue curves) and random beacons (green curves). The main reason is that the beacons obtained via our algorithm generalizes the K-means centers to adapt to the complex manifold structures, and the weight matrix optimized via ℓ_1 regularization (curves with star markers) also reduces unreasonable neighborhood structures by avoiding the artificial parameters, e.g., number of neighboring beacons of each sample "k_n" for LAE (curves with circle markers) [16] and kernel regression (curves with cross markers) [17].

5.5 Discussions

Comprehensive experiments demonstrate that our proposed algorithm is attractive in practical applications. When parameters are set properly, the accuracy is comparable to state-of-the-art [17]. Furthermore, our algorithm offers robustness to sub-optimal parameter and corrupted data, which are essential issues in graph-based semi-supervised learning. Compared with the sample-based algorithms, our proposed algorithm is orders of magnitudes more efficient by omitting the inverse of huge matrices. Besides, we also provide a paradigm to handle statistics shift for time-evolving data by updating the beacon set incrementally.

6 Conclusions

We propose an ℓ_1-Beacon Graph algorithm for graph-based semi-supervised learning, in which the scalable inference is implemented by coupling the design of an informative beacon set and estimation of the sample-to-beacon relationship. Compared with the transductive algorithms [1,2], the proposed algorithm is orders of magnitude more efficient in computation and offers a solution to handle unfamiliar data; moreover, it needs fewer beacons to realize comparable results, since it generalizes the clustered centers by K-means [16,17] and provides more flexible representations. With sparsity and graph smoothness properly harnessed, the algorithm is more robust to corrupted samples. Once unfamiliar samples are encountered, the algorithm is capable of handling novel and unseen data with time-evolving statistics by expanding the beacon set and updating the beacon-related parameters incrementally.

References

1. Chapelle, O., Schlkopf, B., Zien, A.: Semi-Supervised Learning, 1st edn. The MIT Press, Cambrdige (2010)
2. Zhu, X., Goldberg, A.B., Brachman, R., Dietterich, T.: Introduction to Semi-Supervised Learning. Morgan and Claypool Publishers, Cambrdige (2009)
3. Zhu, X.: Semi-supervised learning literature survey. Technical report 1530, Computer Sciences, Carnegie Mellon University (2005)
4. Subramanya, A., Talukdar, P.: Graph-Based Semi-Supervised Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, San Rafael (2014)
5. Blum, A., Chawla, S.: Learning from labeled and unlabeled data using graph mincuts. In: Proceedings of the Eighteenth International Conference on Machine Learning (ICML), pp. 19–26 (2001)
6. Grady, L.: Random walks for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **28**(11), 1768–1783 (2006)
7. Couprie, C., Grady, L., Najman, L., Talbot, H.: Power watershed: a unifying graph-based optimization framework. IEEE Trans. Pattern Anal. Mach. Intell. **33**(7), 1384–1399 (2011)
8. Belkin, M., Niyogi, P., Sindhwani, V.: Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. J. Mach. Learn. Res. **7**, 2399–2434 (2006)
9. Zhu, X., Ghahramani, Z., Lafferty, J.: Semi-supervised learning using Gaussian fields and harmonic functions. In: Twentieth International Conference on Machine Learning (ICML), pp. 912–919 (2003)
10. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Schölkopf, B.: Learning with local and global consistency. In: Advances in Neural Information Processing Systems (NIPS), pp. 321–328. MIT Press (2004)
11. Cheng, B., Yang, J., Yan, S., Fu, Y., Huang, T.S.: Learning with ℓ_1-graph for image analysis. IEEE Trans. Image Process. **19**(4), 858–866 (2010)
12. Yang, Y., Wang, Z., Yang, J., Wang, J., Chang, S., Huang, T.S.: Data clustering by Laplacian regularized ℓ_1-graph. In: Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI), pp. 3148–3149 (2014)
13. Talwalkar, A., Kumar, S., Rowley, H.: IEEE Conference on Large-scale manifold learning. In: Computer Vision and Pattern Recognition (CVPR), pp. 1–8. IEEE (2008)
14. Fergus, R., Weiss, Y., Torralba, A.: Semi-supervised learning in gigantic image collections. In: Advances in Neural Information Processing Systems (NIPS), pp. 522–530 (2009)
15. Dai, D., Van Gool, L.: Ensemble projection for semi-supervised image classification. In: 2013 IEEE International Conference on Computer Vision (CVPR), pp. 2072–2079. IEEE (2013)
16. Liu, W., He, J., Chang, S.F.: Large graph construction for scalable semi-supervised learning. In: Proceedings of the 27th International Conference on Machine Learning (ICML), pp. 679–686 (2010)
17. Chen, X., Cai, D.: Large scale spectral clustering with landmark-based representation. In: The 25th AAAI Conference on Artificial Intelligence (AAAI), pp. 313–318 (2011)
18. Kleinberg, J., Slivkins, A., Wexler, T.: Triangulation and embedding using small sets of beacons. In: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science, pp. 444–453, October 2004

19. Zhang, Z., Ladicky, L., Torr, P., Saffari, A.: Learning anchor planes for classification. In: Advances in Neural Information Processing Systems (NIPS), pp. 1611–1619 (2011)
20. Globerson, A., Roweis, S.: Nightmare at test time: robust learning by feature deletion. In: Proceedings of the 23rd international conference on Machine learning (ICML), pp. 353–360 (2006)
21. Kim, S.J., Koh, K., Lustig, M., Boyd, S., Gorinevsky, D.: An interior-point method for large-scale ℓ_1-1-regularized least squares. IEEE J. Sel. Top. Sig. Process. **1**(4), 606–617 (2007)
22. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online dictionary learning for sparse coding. In: Proceedings of the 26th Annual International Conference on Machine Learning (ICML), pp. 689–696 (2009)
23. Su, H., Yin, Z., Huh, S., Kanade, T.: Cell segmentation in phase contrast microscopy images via semi-supervised classification over optics-related features. Med. Image Anal. **17**, 746–765 (2013)