Instance-Level Label Propagation with Multi-Instance Learning

Qifan Wang, Gal Chechik, Chen Sun and **Bin Shen**
Google Research
Mountain View, CA 94043, US
{wqfcr, gal, chensun, bshen}@google.com

Abstract

Label propagation is a popular semi-supervised learning technique that transfers information from labeled examples to unlabeled examples through a graph. Most label propagation methods construct a graph based on example-to-example similarity, assuming that the resulting graph connects examples that share similar labels. Unfortunately, example-level similarity is sometimes badly defined. For instance, two images may contain two different objects, but have similar overall appearance due to large similar background. In this case, computing similarities based on whole-image would fail propagating information to the right labels. This paper proposes a novel *Instance-Level Label Propagation* (ILLP) approach that integrates label propagation with multi-instance learning. Each example is treated as containing multiple instances, as in the case of an image consisting of multiple regions. We first construct a graph based on instance-level similarity and then simultaneously identify the instances carrying the labels and propagate the labels across instances in the graph. Optimization is based on an iterative Expectation Maximization (EM) algorithm. Experimental results on two benchmark datasets demonstrate the effectiveness of the proposed approach over several state-of-the-art methods.

1 Introduction

Semi-supervised learning is designed to leverage the abundance of unlabeled data to improve subsequent supervised learning tasks. Given a dataset with both labeled and unlabeled data, the goal of semi-supervised learning is to assign labels to the unlabeled examples. Label propagation, (LP) [Zhou *et al.*, 2003; Belkin *et al.*, 2006; Gong *et al.*, 2015], is one major technique in semi-supervised learning [Zhu, 2006]. In label propagation, knowledge about a label is propagated through a similarity graph from labeled examples to unlabeled ones under the assumption that examples (nodes) that are connected in the graph are likely to share the same semantic label.

Label propagation has been widely adopted in many learning tasks, including image classification [Wang and Tsotsos, 2016; Kim *et al.*, 2015], text categorization [Jin *et al.*, 2007; Kim *et al.*, 2009] and information retrieval [Hadiji *et al.*, 2015; Ding and Riloff, 2016]. Most existing label-propagation methods propagate label information across examples through a graph built on example-to-example similarities. These methods treat each example as a single entity, and represent it using one feature vector. In many cases, however, the label of each example may be represented by one or more instances inside the example, rather than the whole entity. For example, when applying label propagation to images (Fig.1), the label of an image often corresponds to one key concept in an image, and that concept is often localized to a limited region of the image. The other parts of the image outside that region could introduce heavy noise when computing similarities based on full images, resulting in poor propagation performance. Similarly, for text categorization, text documents often consist of multiple passages addressing different topics, and computing similarity between two documents based on the overall content is often too coarse.

Some approaches, such as MISSL [Rahmani and Goldman, 2006], directly address the above problem of label propagation over multi-instance examples with two steps. They first assign the concept label of each example to its instances using multi-instance learning (MIL) [Zhou, 2004], and then apply standard LP among the labeled instances. While this approach sometimes performs better than example-level LP, it has two main drawbacks. First, the instance labels obtained from MIL methods are fixed and treated as ground-truth labels in the following label propagation stage. Any labeling error is aggregated and propagated during label propagation. Second, the multi-instance learning and label propagation are performed separately, preventing these two components from benefiting each other and limiting the overall performance.

This paper proposes a novel approach, *Instance-Level Label Propagation*(ILLP), which combines label propagation with multi-instance learning in a single framework. Instead of building a graph over examples, it treats each example as containing multiple instances with each instance as a node, and constructs instance-to-instance similarity graph. Labels are then propagated from instance to instance through this instance-level graph. Specifically, we design a unified

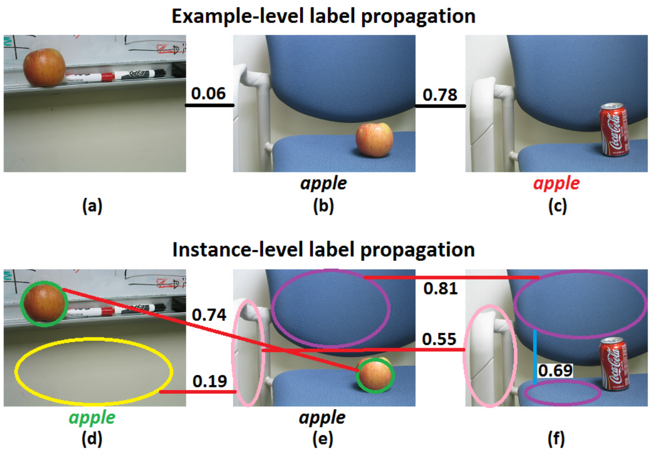

Figure 1: Illustration of example-level label propagation (top row) and proposed instance-level label propagation (bottom row). Examples of Images are from the SIVAL dataset. Top row: image label, '$apple$', gets wrongly propagated from (b) to (c). Bottom row: image (d) obtains the correct label from (e) via ILLP, which also identifies the regions/instances in (d) and (e) that contain '$apple$'. Note that only part of the edges are shown in ILLP for demonstration purpose.

learning framework to incorporate multi-instance learning (MIL) [Dietterich *et al.*, 1997] into label propagation. This joint learning framework enforces multi-instance constraints between example and instances, while at the same time preserves label similarity between instances connected in the graph. An iterative Expectation Maximization (EM) algorithm is then proposed to solve the optimization problem based on this learning framework. Experimental results on two benchmarks demonstrate the advantage of ILLP over several baseline methods.

2 Related Work

2.1 Label Propagation

In many real-world applications, large amounts of unlabeled data are available, but labeling all data is expensive or impossible. Semi-supervised learning [Chapelle *et al.*, 2010] has been proposed to leverage the information from unlabeled data for achieving better learning results. Label propagation is a popular graph-based semi-supervised method, which incorporates the knowledge from unlabeled data through propagating the labels from labeled data.

Label propagation was introduced in [Zhu and Ghahramani, 2002; Zhou *et al.*, 2003]. They create a similarity graph among data examples and allow every example iteratively spread its label information to its neighbors until a global stable state is achieved. A linear neighborhood propagation (LNP) method is presented in [Wang and Zhang, 2006], which assumes that each data example can be linearly reconstructed from its neighborhood. The labels are then propagate from the labeled examples to the whole dataset using these linear neighborhoods with sufficient smoothness. A non-metric label propagation method [Zhang and Zhou, 2009] is designed that decomposes the non-metric distance matrix first and then conducts a joint label propagation on the joint graph. Manifold regularization is employed in [Karasuyama and Mamitsuka, 2013], which is built upon a Laplacian graph for label propagation, to capture the manifold structure of labeled and unlabeled examples. Several recent work [Gong *et al.*, 2016; Kang *et al.*, 2006] consider multi-label learning problem with label propagation, which simultaneously co-propagates multiple labels by explicitly modeling the correlations between labels in an efficient manner. More recently, an efficient label propagation algorithm is developed in [Fujiwara and Irie, 2014] which achieves the optimal solution much faster.

2.2 Multi-Instance Learning

Multi-Instance Learning (MIL) [Dietterich *et al.*, 1997; Maron and Lozano-Pérez, 1997; Strelow *et al.*, 2016] aims to solve the label ambiguity problem, namely, assigning a label that is known at the example level to one or more instances in that example. In this way, the features for the desired local object in each example will be less likely affected by its irrelevant parts, and therefore the learned model can be more accurate. Most MIL algorithms can be divided into two groups, generative models [Maron and Lozano-Pérez, 1997; Zhang and Goldman, 2001] and discriminative models [Andrews *et al.*, 2002; Chen *et al.*, 2006; Wang *et al.*, 2012]. A comprehensive survey of multi-instance learning is summarized in [Zhou, 2004; Amores, 2013]. The relation of MIL and semi-supervised learning has been explored by several authors. MIL problem is reformulated as a specific semi-supervised problem in [Zhou and Xu, 2007]. Jia et al. propose to boost the performance of MIL using unlabeled data by incorporating a graph $Laplacian$ term in to MIL framework [Jia and Zhang, 2008]. A direct combination of MIL and label propagation method is proposed in [Rahmani and Goldman, 2006], sharing the same goal with this work. However, as aforementioned, this approach models MIL and semi-supervised learning in two separate steps, potentially leading to poor performance if the generated labels are noisy. This is discussed further in the experiment section.

3 Preliminary

In this section, we briefly review the label propagation approach described in [Zhou *et al.*, 2003; Fujiwara and Irie, 2014] that applied to example-level graphs. Let $X = \{x_1, x_2, \ldots, x_m, x_{m+1}, \ldots, x_n\}$ represents a set of data examples, where the first m data examples are labeled with $\{y_1, y_2, \ldots, y_m\}$ and $y_i \in \{l_1, l_2, \ldots, l_c\}$. Here c is the size of the vocabulary of labels (Note that this work solves multi-class single-label problem). The remaining data examples are unlabeled. A graph $G = \{V, E\}$ is constructed over the data examples $V = \{x_i\}$, and the edges of the graph are assigned weights $\boldsymbol{W} = \{W_{ij}\}$ that are given in advance. Weights could be created using k-nearest neighbors (k-NN)[von Luxburg, 2007]. Let $\boldsymbol{Y} \in \{0, 1\}^{n \times c}$ denote the matrix of labels, where $Y_{ip} = 1$ if example x_i is labeled as $y_i = l_p$ and $Y_{ip} = 0$ otherwise. Let $\boldsymbol{S} \in \mathbb{R}^{n \times c}$ denote a matrix of classification scores. We denote by $\boldsymbol{S_i}$ and $\boldsymbol{Y_i}$ the i^{th} row vector of $\boldsymbol{S}$

and $\boldsymbol{Y}$ respectively. In label propagation, the classification scores are defined as the optimal solution to the following cost minimization problem:

$$\boldsymbol{S} = \arg\min_{\boldsymbol{S}} \sum_{i=1}^{n} \|\boldsymbol{S_i} - \boldsymbol{Y_i}\|^2 + \alpha \sum_{i,j=1}^{n} W_{ij} \| \frac{\boldsymbol{S_i}}{\sqrt{D_{ii}}} - \frac{\boldsymbol{S_j}}{\sqrt{D_{jj}}} \|^2 \tag{1}$$

where $\boldsymbol{D}$ is a diagonal matrix with $D_{ii} = \sum_{j=1}^{n} W_{ij}$, and α is a hyper-parameter that balances the two terms. The first term in the cost function represents the fitting constraint, which ensures that the solution is close to the initial label assignment. The second term corresponds to the smoothness constraint, ensuring that the solution assigns similar scores to nodes that are connected by large weights. Minimizing the cost function of Eqn. 1 has a closed-form optimal solution:

$$\boldsymbol{S} = ((\alpha + 1)\boldsymbol{I} - \alpha \boldsymbol{L})^{-1} \boldsymbol{Y} \tag{2}$$

where $\boldsymbol{I}$ is an identity matrix of size $n \times n$ and $\boldsymbol{L} = \boldsymbol{D}^{-1/2} \boldsymbol{W} \boldsymbol{D}^{-1/2}$. The final label y_i of example x_i is obtained by selecting the label with the highest score $y_i = \text{argmax}_p \, S_{ip}$.

4 ILLP: Instance-Level Label Propagation with Multi-Instance Learning

4.1 Problem Setting

We adapt the above notation for ILLP. In ILLP, each data example x_i is represented by n_i instances, namely, $x_i = \{x_{i1} \ldots, x_{iu}, \ldots, x_{in_i}\}$ where x_{iu} is the u^{th} instance of example x_i. The details of generating data instances are given in the experiment section. The total number of instance is denoted by $N = \sum_i n_i$. The goal of ILLP is to assign labels to the unlabeled examples, that is, obtain $\{y_i | i = m+1, m+2, \ldots, n\}$, while at the same time identify the instances that contain the labels.

4.2 Instance-Level Graph Construction

Unlike traditional label propagation, in IILP the similarity graph $G = \{V, E\}$ is constructed over instances instead of over examples. Namely, each instance x_{iu} is treated as a node in V, and the set of weighted edges are constructed among instances. Numerous ways have been suggested for defining the similarity matrix $\boldsymbol{W} \in \mathbb{R}^{N \times N}$. For example, in spectral hashing [Weiss *et al.*, 2008], the authors used the global similarity structure of all data pairs, while in [Fujiwara and Irie, 2014], the local similarity structure, based on k-nearest-neighborhood is used. Here we adopt the local similarity. Specifically, we assume that one can compute a Euclidean distance between pairs of samples, and set the weight between instances x_{iu} and x_{jv} to decay exponentially with the square of the distance, like in a Gaussian function:

$$\boldsymbol{W}_{iu,jv} = \begin{cases} e^{-\frac{\|x_{iu} - x_{jv}\|^2}{\sigma_{ij}^2}} & , if \; x_{iu} \in N_k(x_{jv}) \; or \; x_{jv} \in N_k(x_{iu}) \\ 0, & otherwise \end{cases} \tag{3}$$

The variance σ_{ij} is determined automatically by local scaling [Zelnik-Manor and Perona, 2004], and $N_k(x)$ represents the set of k-nearest-neighbors of data instance x. Note that the k-NN scheme indicates that the number of edges is $O(Nk)$ with $k \ll N$ in practical and the graph is symmetric.

4.3 Proposed Formulation

Similar to example-level label propagation, a set of variables $\boldsymbol{S} \in \mathbb{R}^{N \times c}$ is introduced in ILLP representing the classification scores for all N instances and c classes. Recall that one of the main challenges in designing ILLP is label ambiguity, thus we introduce a set of latent variables y_{iu} to indicate the label on instance x_{iu}. Let $\boldsymbol{Y} \in \{0,1\}^{N \times c}$ denote the unknown label matrix over all instances, where Y_{iu}^{p} corresponds to the iu^{th} row and p^{th} column of $\boldsymbol{Y}$, and $Y_{iu}^{p} = 1$ if instance x_{iu} has label $y_{iu} = l_p$ and $Y_{iu}^{p} = 0$ otherwise. Let $\boldsymbol{S_{iu}}$ and $\boldsymbol{Y_{iu}}$ be the row vectors of $\boldsymbol{S}$ and $\boldsymbol{Y}$ respectively. In ILLP, the labeling score and instance label are jointly optimized in the following formulation:

$$\begin{aligned} \min_{\boldsymbol{S}, \boldsymbol{Y}} = & \sum_{i=1}^{n} \sum_{u=1}^{n_i} \|\boldsymbol{S_{iu}} - \boldsymbol{Y_{iu}}\|^2 \\ & + \alpha \sum_{i}^{n} \sum_{u,v} W_{iu,iv} \| \frac{\boldsymbol{S_{iu}}}{\sqrt{D_{iu}}} - \frac{\boldsymbol{S_{iv}}}{\sqrt{D_{iv}}} \|^2 \\ & + \beta \sum_{iu,jv,i \neq j} W_{iu,jv} \| \frac{\boldsymbol{S_{iu}}}{\sqrt{D_{iu}}} - \frac{\boldsymbol{S_{jv}}}{\sqrt{D_{jv}}} \|^2 \\ & s.t. \quad \boldsymbol{Y_{iu}} \in \{0,1\}^c \\ & \sum_{u=1}^{n_i} Y_{iu}^{p} \geq 1, \; \sum_{u=1}^{n_i} Y_{iu}^{q} = 0 \quad \forall y_i = l_p, q \neq p. \end{aligned} \tag{4}$$

Here, $\boldsymbol{D} \in \mathbb{R}^{N \times N}$ is a diagonal matrix computed from $\boldsymbol{W}$ with $D_{iu,iu} = \sum_{j=1}^{N} W_{iu,j}$. The trade-off parameters α and β balance the three components of the loss. The first term in the cost function represents the fitting criteria at instance level, which ensures that good classification should be consistent with the label assignment. The second term corresponds to the similarity preservation for instances from the same example, while the third term preserves the similarity of instances from different examples. These similarity preservation terms enable that classification scores should not vary too much between similar instances. The reason of adapting two similarity terms in ILLP formulation is that we want to distinguish edges that are of different types in the graph. The first constraint in the objective function is the binary constraint on the instance labels. The second constraint represents the multi-instance constraint imposed between example and instances. Essentially, this constraint can be interpreted as: for example x_i with label l_p, at least one of its instance should contain label l_p ($\sum_{u=1}^{n_i} Y_{iu}^{p} \geq 1$), while none of its instance contains other labels (since l_p is the only label assigned to x_i[1]). Note that the second constraint is only on labeled examples $\{x_1, \ldots, x_m\}$. It is clear that the example-level label propagation formulation in Eqn.1 is a special case of the above ILLP formulation by treating each

[1] This work focuses on multi-class single-label problem in label propagation. However, for multi-class multi-label problem, this constraint could be relaxed.

example as its only instance (i.e., $n_i = 1$ for all i) and ignoring the second similarity preservation term.

There are three major differences between ILLP formulation in Eqn.4 and previous modeling in Eqn.1. First, the instance label matrix $\boldsymbol{Y}$ in ILLP is not known, but needs to be inferred together with the instance classification/labeling score matrix $\boldsymbol{S}$. Actually, as we will see in the optimization section, this label matrix keeps updating during iterations. Second, the multi-instance constraint and binary constraint imposed on label matrix makes the joint optimization problem intractable with no closed-form solution. Third, the ILLP formulation treats edges that connecting instances between examples and within same example differently, and thus is capable of balancing the weights between the two terms.

4.4 Optimization

Directly minimizing the objective in Eqn.4 is intractable, since model parameters $\boldsymbol{S}$ and $\boldsymbol{Y}$ are coupled together with binary constraints, resulting in a non-convex non-smooth optimization problem. An Expectation-Maximization (EM) like iterative method is employed to solve this problem. In particular, we optimize the objective function with respect to model parameters $\boldsymbol{S}$ and $\boldsymbol{Y}$ alternatively by the following two steps.

E-Step: Fix labeling scores $\boldsymbol{S}$, update instance labels $\boldsymbol{Y}$. Given the labeling score matrix $\boldsymbol{S}$, the optimization problem becomes the following n sub-problems for n examples:

$$\begin{aligned}\boldsymbol{Y_{iu}^*} = \underset{\boldsymbol{Y_{iu}}, 1\leq u\leq n_i}{\arg\min} \sum_{u=1}^{n_i} \|\boldsymbol{S_{iu}} - \boldsymbol{Y_{iu}}\|^2 \quad & i = \{1, 2, \ldots, n\} \\ s.t. \quad \boldsymbol{Y_{iu}} \in \{0,1\}^c & \\ \sum_{u=1}^{n_i} Y_{iu}^p \geq 1, \sum_{u=1}^{n_i} Y_{iu}^q = 0 \quad & \forall y_i = l_p, q \neq p\end{aligned} \tag{5}$$

For the unlabeled examples $\{x_{m+1}, \ldots, x_n\}$, note that the second multi-instance constraint in Eqn.5 will not apply. Thus the optimal solution on unlabeled data can be obtained by directly binarizing $\boldsymbol{S_{iu}}$, i.e., $\boldsymbol{Y_{iu}}^* = sgn(\boldsymbol{S_{iu}} - \frac{1}{2}\boldsymbol{1_c})$ (for $m+1 \leq i \leq n$), where $\boldsymbol{1_c}$ is a c dimension vector of all ones and sgn is the sign function.

For the labeled example x_i ($1 \leq i \leq m$), the optimal solution to Eqn.5 can be obtained by the following process. First of all, due to constraint $\sum_{u=1}^{n_i} Y_{iu}^q = 0$, it is clear to set ${Y_{iu}^q}^*$ to 0 for all instances of x_i, where $q \neq p$. Then assigning $\hat{Y_{iu}^p} = sgn(S_{iu}^p - 0.5)$ for all instances x_{iu} by relaxing the constraint $\sum_{u=1}^{n_i} Y_{iu}^p \geq 1$. If the solution $\hat{Y_{iu}^p}$ already satisfies the relaxed constraint, it can be shown that the optimal solution of $\boldsymbol{Y_{iu}}^*$ to Eqn.5 is the combination of $\hat{Y_{iu}^p}$ and ${Y_{iu}^q}^*$. Otherwise, if $\sum_{u=1}^{n_i} \hat{Y_{iu}^p} < 1$ (i.e., $\hat{Y_{iu}^p} = 0$ for all u), which means none of the instances is assigned with the example label l_p. In this case, the optimal solution can be achieved by assigning label l_p to the instance with the highest labeling score on l_p, i.e., set $Y_{iu^*}^p = 1$, where $u^* = \text{argmax}_u\, \boldsymbol{S_{iu}}$.

The E-step can also be viewed as instance identification/selection in our ILLP optimization. Similar instance selection method is also adopted in some MIL research [Andrews *et al.*, 2002; Wang *et al.*, 2014].

M-step: Fix instance labels $\boldsymbol{Y}$, obtain the optimal classification scores $\boldsymbol{S}$. Given $\boldsymbol{Y}$, the objective function can be written as:

$$\begin{aligned}\boldsymbol{S}^* = \underset{\boldsymbol{S}}{\arg\min} & \sum_{i=1}^{n}\sum_{u=1}^{n_i} \|\boldsymbol{S_{iu}} - \boldsymbol{Y_{iu}}\|^2 \\ +\alpha & \sum_{i}^{n}\sum_{u,v} W_{iu,iv} \|\frac{\boldsymbol{S_{iu}}}{\sqrt{D_{iu}}} - \frac{\boldsymbol{S_{iv}}}{\sqrt{D_{iv}}}\|^2 \\ +\beta & \sum_{iu,jv,i\neq j} W_{iu,jv} \|\frac{\boldsymbol{S_{iu}}}{\sqrt{D_{iu}}} - \frac{\boldsymbol{S_{jv}}}{\sqrt{D_{jv}}}\|^2\end{aligned} \tag{6}$$

The above sub-problem is less complicated than the problem in Eqn.4 and it is differentiable with respect to $\boldsymbol{S}$. By taking the derivative of the above objective and set it to 0, an optimal solution of $\boldsymbol{S}$ can be obtained. We omit the derivation and directly present the result as follows:

$$\boldsymbol{S}^* = (\boldsymbol{I} - \boldsymbol{D}^{-1/2}(\alpha \boldsymbol{W^I} + \beta \boldsymbol{W^B})\boldsymbol{D}^{-1/2})^{-1}\boldsymbol{Y} \tag{7}$$

where matrix $\boldsymbol{W^I}$ is the sub-matrix of $\boldsymbol{W}$ with elements representing instances similarity from same example, i.e., $\boldsymbol{W^I} = \{W_{iu,jv} | i = j\}$. Matrix $\boldsymbol{W^B}$ is the sub-matrix of $\boldsymbol{W}$ containing instances similarity between different examples, $\boldsymbol{W^B} = \{W_{iu,jv} | i \neq j\}$. The M-step in the ILLP optimization is essentially instance label propagation process, which finds the global optimal solution for the labeling scores $\boldsymbol{S}$ on all instances by propagating the instance labels from the label matrix $\boldsymbol{Y}$. The above EM steps are performed alternatively until convergence. In our implementation, the EM algorithm is terminated if the number of iterations reaches 100, or if labels do not change during two consecutive iterations. The final label of example x_i is then obtained by picking the label with the highest labeling score in $\boldsymbol{S_{iu}}$ for all instances x_{iu}, i.e., $y_i = \text{argmax}_p\, S_{iu}^p$. The full optimization algorithm is described in Algorithm 1, and its convergence is guaranteed by the following theorem:

Theorem 1. *The iterative EM algorithm described in Algorithm 1 terminates in a finite number of iterations.*

Proof. First, it is clear that the objective defined in Eqn.4 will decrease consistently during the EM iterations. Note that the E-step is essentially assigning labels to instances, and there are only finite number of possible assignments. At each EM iteration, the instance label assignment differs from previous assignments, otherwise, the objective would remain unchanged and not decrease. Therefore, the EM algorithm terminates in a finite number of steps. □

4.5 Analysis

This section provides analysis on the training cost of the EM algorithm. In the E-step, we solve n sub-problems to obtain the instance labeling scores, one for each example, through Eqn.5, where the time complexity for computing the

Algorithm 1 Instance-Level Label Propagation with Multi-Instance Learning (ILLP)

```
Input: Labeled examples ${(x_{iu}, y_i)|1 \le i \le m}$ with $y_i \in {l_1, l_2, \dots, l_c}$. Unlabeled examples ${x_{iu}|m+1 \le i \le n}$ and trade-off parameters $\alpha$ and $\beta$.
Output: Instance classification scores $\boldsymbol{S}$, instance label matrix $\boldsymbol{Y}$ and example labels $y_i$.
Construct instance level similarity graph $\boldsymbol{W}$ by Eqn.3.
Initialize model parameters $\boldsymbol{S}$ and $\boldsymbol{Y}$.
repeat
    E-step: Instance identification from Eqn.5
        Set ${Y^{q}_{iu}}^* = 0, {Y^{p}_{iu}}^* = sgn(S^p_{iu} - 0.5),\ 1 \le i \le m$
            If $\sum_{u=1}^{n_i} {Y^{p}_{iu}}^* < 1$, reset $Y^{p}_{iu^*} = 1$ where $u^* =$
            $\mathrm{argmax}_u\, \boldsymbol{S_{iu}}$
        Set $\boldsymbol{Y_{iu}}^* = sgn(\boldsymbol{S_{iu}} - \frac{1}{2}\boldsymbol{1_c}), \quad m+1 \le i \le n$
    M-step: Instance label propagation by Eqn.7
        $\boldsymbol{S}^* = (\boldsymbol{I} - \boldsymbol{D}^{-1/2}(\alpha \boldsymbol{W^I} + \beta \boldsymbol{W^B})\boldsymbol{D}^{-1/2})^{-1}\boldsymbol{Y}$
        Optimize using sparse power method.
until EM converges
Obtain example label $y_i = \mathrm{argmax}_p\, S^p_{iu}$, for all $u$.
```

binary labels is proportion to the total number of instances and total number of class labels, $O(Nc)$. For the M-step, directly applying Eqn.7 involves inverting an matrix, with a computation cost of $O(N^3)$. However, recall that the matrix needs to be inverted in Eqn.7 is a sparse symmetric matrix with $O(Nk)$ non-zero elements. (k is the number of neighbors). Therefore in our implementation, we employ the power method [Golub and Van Loan, 2012] to efficiently obtain the optimal solution without matrix inversion. The computation complexity then becomes $O(Nkt)$ where t is the number of iteration in the power method. Moreover, we found in our experiments that the EM usually converges in less than 80 iterations. Thus, the total time complexity of the learning algorithm is bounded by $O(Nkt + Nc)$, which scales linearly with N (given $N \gg t$ and $N \gg k$).

In the experiments, we also found out that the convergence rate of our learning algorithm is sensitive to the initialization of $\boldsymbol{Y}$. In other words, a better initial instance labeling will result in much faster convergence speed. In our implementation, we adopt the instance selection method in MIL [Fu and Robles-Kelly, 2009] to initialize the instance label matrix to further accelerate the learning algorithm.

5 Experimental Results

5.1 Datasets and Setting

The proposed ILLP approach is evaluated with three configurations of experiments on two benchmarks: an image dataset **SIVAL**[2] and a text corpus **Reuters** ($Reuters21578$)[3]. The **SIVAL** benchmark is a widely used MIL dataset. It contains 25 image categories with 60 images in each category. Each image is segmented into multiple regions [Shi and Malik, 2000], which are treated as instances. The total number of instance is around $45k$. A set of low-level features from each segment is extracted to represent an instance, including color histogram, color moment, region size, wavelet texture and shape [Wang *et al.*, 2012]. **Reuters** is a benchmark dataset from Reuters newswire in 1987. It has 135 categories/labels, with 21578 documents. We select the five largest categories in our experiments. Since this work focuses on multi-class single-label problem, we remove those documents that have more than one label, resulting in 5346 documents. Similar to [Andrews *et al.*, 2002], we treat each document as an example and use fixed-length passages as instances, resulting in total $137k$ instances. After removing stopwords and stemming, tf-idf [Zhang *et al.*, 2013] features are extracted.

	SIVAL	Reuters
ILLP	**0.823 ± 0.014**	**0.786 ± 0.011**
LP	0.750 ± 0.015	0.742 ± 0.013
M^{3}IL	0.763 ± 0.012	0.737 ± 0.016
M^{3}IL + LP	0.781 ± 0.011	0.759 ± 0.016
MISSL	0.722 ± 0.023	0.717 ± 0.024

Table 1: Average AUC on two benchmarks for all compared methods.

In each experiment, we randomly partition the examples in each category into two splits to form the labeled and unlabeled sets. The trade-off parameters α and β are tuned using five-fold cross-validation. We set the number of neighbors k to 8 to construct the k-NN graph. We quantify the quality of the models using the average area under the ROC curve (AUC) measure on unlabeled data.

5.2 Evaluation of Different Algorithms

We compare the proposed ILLP approach with four methods: **(1) LP:** The example-level label propagation method by [Zhou *et al.*, 2003; Fujiwara and Irie, 2014]. Example-level features are used to construct the similarity graph. We used the same number of neighbors as in ILLP $k = 8$. The hyper-parameter α is tuned with five-fold cross validation. **(2) M^{3}IL:** The state-of-the-art MIL approach by [Wang *et al.*, 2012], which demonstrates superior performance over many other MIL methods. The number of clusters is set to be 3. For other parameters λ and β, we used the values in the original implementation provided by the authors. **(3) MISSL:** The method of [Rahmani and Goldman, 2006] that directly combines MIL [Maron and Lozano-Pérez, 1997] with label propagation. (The code is available from http://www.cs.cmu.edu/~juny/MILL/). MISSL first applies MIL method on the labeled data to obtain instance labels. Then, labels are propagated across the instance graph with the pre-generated instance labels. **(4) M^{3}IL+LP:** Similar to MISSL, it is a combination approach of M^{3}IL and LP that applies multi-instance learning and label propagation separately. The same k is used in MISSL and M^{3}IL+LP as in ILLP. The first two methods, M^{3}IL and LP, can be viewed as two individual components of our ILLP, while M^{3}IL+LP and MISSL are methods that sequentially apply these two components.

The average AUC of all methods are reported in Table 1. ILLP clearly outperforms all compared baselines on both

[2] http://www.cs.wustl.edu/~sg/multi-inst-data/

[3] http://www.daviddlewis.com/resources/testcollections/

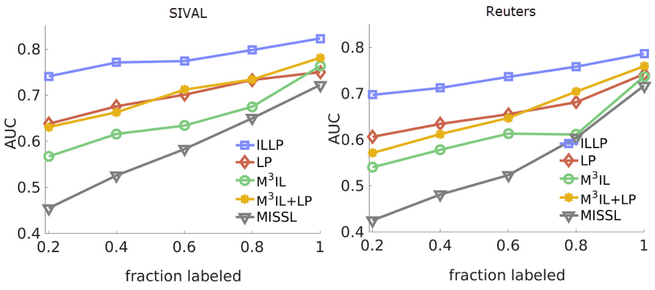

Figure 2: Average AUC results on two benchmarks by varying ration of training (labeled) examples.

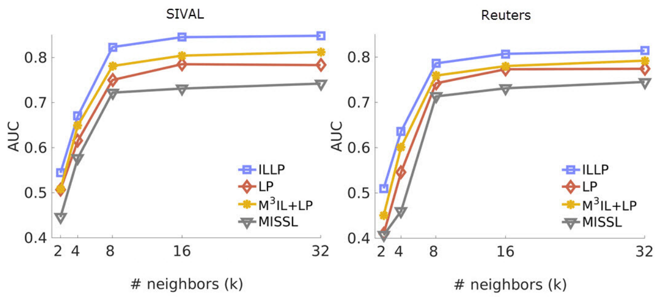

Figure 3: Average AUC results of different number of k on two benchmarks.

datasets. As expected, ILLP outperforms both LP and M^{3}IL methods which do not combine LP with MIL. Compared to example-based LP, which is bound to use ambiguous example-to-example similarity, ILLP clearly benefits from multi-instance modeling and transfers label information accurately from instance to instance. Compared with M^{3}IL, which does not leverage knowledge from the unlabeled data, ILLP can generate high accuracy labels. ILLP also outperforms the two combination methods, M^{3}IL+LP and MISSL, which is consistent with our expectation. The reason is that both these methods apply MIL and label propagation separately in a sequential manner, while ILLP jointly models both two parts in a unified framework and keeps updating the instance labels and classification scores alternatively during the optimization. Furthermore, the knowledge contained in unlabeled data is not used by the MIL methods in the stage of labeling instances. In other words, any mistake made by MIL methods during the instance labeling stage is aggregated and propagated during label propagation.

5.3 The Effect of Labeling Ratio

To evaluate the effectiveness of the proposed ILLP approach as a function of the fraction of labeled examples, we progressively increase the number of labeled examples by varying the *labeling ratio*, namely, the number of labeled example divided by number of unlabeled example, in the set $\{0.2, 0.4, 0.6, 0.8, 1\}$ and compare ILLP with all the other baseline methods on the two benchmark datasets. Fig.2 depicts the average AUC as a function of labeling ratio showing that ILLP outperforms all compared methods on different training ratios. It can be observed from Fig.2 that the performance of label propagation methods, ILLP and LP, suffers less with small ratio of labeled data than M^{3}IL+LP and MISSL methods. The reason is that the classifiers learned by MIL methods generate more and more mis-classifications on instances with the decreasing of training/labeled data. These errors are then get accumulated and transferred to the unlabeled data during label propagation, resulting in even poorer performance. While label propagation methods utilize both labeled and unlabeled data to achieve global optimal labeling scores. However, our ILLP consistently outperforms LP on different training ratios. We attribute this to the advantage of incorporating multi-instance learning, which is capable of assigning accurate labels to instances, and performing better label propagation.

5.4 The Effect of Number of Neighbors k

The performance of label propagation highly depends on the quality of the similarity graph, on which labels are propagated. To evaluate the effectiveness of the proposed ILLP, we test different number of neighbors, by varying k from the set $\{2, 4, 8, 16, 32\}$, and constructing the similarity graph for each k value . Fig.3 compares the AUC as a function of neighborhood size, showing that ILLP outperforms the competing methods (M^{3}IL is not available here since it does not contain LP part) on all k values. As we can see in the figure, the AUC value of ILLP gets saturated when the number of k approaches around 16 on both datasets, which is consistent with our expectation. Since with the increasing number of neighbors, more redundant information will be represented in the graph, resulting in limited performance boost. Similar patterns are also observed in the figure for other methods. This is also the reason why we set k to 8 in our previous experiments.

6 Conclusion

This paper proposes a novel approach of Instance-Level Label Propagation (ILLP) with multi-instance learning. The new method constructs instance-level similarity graph instead of example-level graph for label propagation, which better captures the semantic similarity between examples. A unified learning framework is developed which enables simultaneously label propagation between instances and label identification within example. The optimization problem is solved by an efficient iterative EM algorithm. Experimental results on two datasets demonstrate the advantage of the proposed ILLP approach against several baselines. In future, we plan to develop theoretical analysis of the generalization error of the proposed learning algorithm. We also plan to extend our ILLP to the multi-class multi-label scenario.

Acknowledgments

We would like to thank our colleagues Rahul Sukthankar and Tom Duerig for their insightful suggestions on this work, as well as revising the paper.

References

[Amores, 2013] Jaume Amores. Multiple instance classification: Review, taxonomy and comparative study. *Artif. Intell.*, 201:81–105, 2013.

[Andrews *et al.*, 2002] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. Support vector machines for multiple-instance learning. In *NIPS*, pages 561–568, 2002.

[Belkin *et al.*, 2006] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7:2399–2434, 2006.

[Chapelle *et al.*, 2010] Olivier Chapelle, Bernhard Schlkopf, and Alexander Zien. *Semi-Supervised Learning*. The MIT Press, 1st edition, 2010.

[Chen *et al.*, 2006] Yixin Chen, Jinbo Bi, and James Ze Wang. Miles: Multiple-instance learning via embedded instance selection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(12):1931–1947, 2006.

[Dietterich *et al.*, 1997] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artif. Intell.*, 89(1-2):31–71, 1997.

[Ding and Riloff, 2016] Haibo Ding and Ellen Riloff. Acquiring knowledge of affective events from blogs using label propagation. In *AAAI*, pages 2935–2942, 2016.

[Fu and Robles-Kelly, 2009] Zhouyu Fu and Antonio Robles-Kelly. An instance selection approach to multiple instance learning. In *CVPR*, pages 911–918, 2009.

[Fujiwara and Irie, 2014] Yasuhiro Fujiwara and Go Irie. Efficient label propagation. In *ICML*, pages 784–792, 2014.

[Golub and Van Loan, 2012] Gene H Golub and Charles F Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

[Gong *et al.*, 2015] Chen Gong, Dacheng Tao, Keren Fu, and Jie Yang. Fick's law assisted propagation for semisupervised learning. *IEEE Trans. Neural Netw. Learning Syst.*, 26(9):2148–2162, 2015.

[Gong *et al.*, 2016] Chen Gong, Dacheng Tao, Jie Yang, and Wei Liu. Teaching-to-learn and learning-to-teach for multi-label propagation. In *AAAI*, pages 1610–1616, 2016.

[Hadiji *et al.*, 2015] Fabian Hadiji, Martin Mladenov, Christian Bauckhage, and Kristian Kersting. Computer science on the move: Inferring migration regularities from the web via compressed label propagation. In *IJCAI*, pages 171–177, 2015.

[Jia and Zhang, 2008] Yangqing Jia and Changshui Zhang. Instance-level semisupervised multiple instance learning. In *AAAI*, pages 640–645, 2008.

[Jin *et al.*, 2007] Rong Jin, Ming Wu, and Rahul Sukthankar. Semi-supervised collaborative text classification. In *ECML*, pages 600–607, 2007.

[Kang *et al.*, 2006] Feng Kang, Rong Jin, and Rahul Sukthankar. Correlated label propagation with application to multi-label learning. In *CVPR*, pages 1719–1726, 2006.

[Karasuyama and Mamitsuka, 2013] Masayuki Karasuyama and Hiroshi Mamitsuka. Manifold-based similarity adaptation for label propagation. In *NIPS*, pages 1547–1555, 2013.

[Kim *et al.*, 2009] Soo-Min Kim, Patrick Pantel, Lei Duan, and Scott Gaffney. Improving web page classification by label-propagation over click graphs. In *CIKM*, pages 1077–1086, 2009.

[Kim *et al.*, 2015] Kwang In Kim, James Tompkin, Hanspeter Pfister, and Christian Theobalt. Context-guided diffusion for label propagation on graphs. In *ICCV*, pages 2776–2784, 2015.

[Maron and Lozano-Pérez, 1997] Oded Maron and Tomás Lozano-Pérez. A framework for multiple-instance learning. In *NIPS*, 1997.

[Rahmani and Goldman, 2006] Rouhollah Rahmani and Sally A. Goldman. MISSL: multiple-instance semi-supervised learning. In *ICML*, pages 705–712, 2006.

[Shi and Malik, 2000] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, 2000.

[Strelow *et al.*, 2016] Dennis Strelow, Qifan Wang, Luo Si, and Anders Eriksson. General, nested, and constrained wiberg minimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(9):1803–1815, 2016.

[von Luxburg, 2007] Ulrike von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[Wang and Tsotsos, 2016] Bo Wang and John K. Tsotsos. Dynamic label propagation for semi-supervised multi-class multi-label classification. *Pattern Recognition*, 52:75–84, 2016.

[Wang and Zhang, 2006] Fei Wang and Changshui Zhang. Label propagation through linear neighborhoods. In *ICML*, pages 985–992, 2006.

[Wang *et al.*, 2012] Qifan Wang, Luo Si, and Dan Zhang. A discriminative data-dependent mixture-model approach for multiple instance learning in image classification. In *ECCV (4)*, pages 660–673, 2012.

[Wang *et al.*, 2014] Qifan Wang, Lingyun Ruan, and Luo Si. Adaptive knowledge transfer for multiple instance learning in image classification. In *AAAI*, pages 1334–1340, 2014.

[Weiss *et al.*, 2008] Yair Weiss, Antonio Torralba, and Robert Fergus. Spectral hashing. In *NIPS*, pages 1753–1760, 2008.

[Zelnik-Manor and Perona, 2004] Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. In *NIPS*, 2004.

[Zhang and Goldman, 2001] Qi Zhang and Sally A. Goldman. Em-dd: An improved multiple-instance learning technique. In *NIPS*, pages 1073–1080, 2001.

[Zhang and Zhou, 2009] Yin Zhang and Zhi-Hua Zhou. Non-metric label propagation. In *IJCAI*, pages 1357–1362, 2009.

[Zhang *et al.*, 2013] Dan Zhang, Jingrui He, and Richard D. Lawrence. M2ls: Multi-instance learning from multiple information sources. In *KDD*, 2013.

[Zhou and Xu, 2007] Zhi-Hua Zhou and Jun-Ming Xu. On the relation between multi-instance learning and semi-supervised learning. In *ICML*, pages 1167–1174, 2007.

[Zhou *et al.*, 2003] Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In *NIPS*, pages 321–328, 2003.

[Zhou, 2004] Zhi-Hua Zhou. Multi-Instance learning: A survey. Technical report, 2004.

[Zhu and Ghahramani, 2002] Xiaojin Zhu and Zoubin Ghahramani. Learning from labeled and unlabeled data with label propagation. Technical report, 2002.

[Zhu, 2006] Xiaojin Zhu. Semi-supervised learning literature survey, 2006.