Conditional Restricted Boltzmann Machines for Multi-label Learning with Incomplete Labels

Xin Li and **Feipeng Zhao** and **Yuhong Guo**
Department of Computer and Information Sciences
Temple University, Philadelphia, PA 19122, USA
{xinli, feipeng.zhao, yuhong}@temple.edu

Abstract

Standard multi-label learning methods assume fully labeled training data. This assumption however is impractical in many application domains where labels are difficult to collect and missing labels are prevalent. In this paper, we develop a novel conditional restricted Boltzmann machine model to address multi-label learning with incomplete labels. It uses a restricted Boltzmann machine to capture the high-order label dependence relationships in the output space, aiming to enhance the capacity of recovering missing labels and learning high quality multi-label prediction models. Moreover, it also incorporates label co-occurrence information retrieved from auxiliary resources as prior knowledge. We perform model training by maximizing the regularized marginal conditional likelihood of the label vectors given the input features, and develop a Viterbi style EM algorithm to solve the induced optimization problem. The proposed approach is evaluated on four real word multi-label data sets by comparing to a number of state-of-the-art methods. The experimental results show it outperforms all the other comparison methods across the applied data sets.

1 Introduction

Multi-label learning is critical in many real world application domains where an instances can be associated with multiple (possibly related) label concepts simultaneously. For example, one image can contain multiple objects, such as "person", "car", and "road", and hence belong to multiple label categories. In the past, multi-label learning has attracted intensive attention and many algorithms have been developed in the literature, including graphical model based methods [9, 10, 29]. These methods typically assume that all the instances in the training data have complete labels and learn a prediction model to map the instances from the input feature space to the given label vectors. The assumption of complete labels however is impractical in many real world application domains, where it is difficult to acquire a complete set of true label assignments from the annotators. For example, annotators for images or articles may only provide the most obvious labels they found while ignoring the ambiguous labels or the label concepts they are not familiar with. Ignoring the missing labels in the training data however can significantly degrade the performance of the learned multi-label classification model, since it will build negative prediction patterns between the input instances and their missing labels and further propagate the mistakes into the prediction phase on new data. This raises the significant challenge of multi-label learning with incomplete labels.

A key for tackling multi-label learning with incomplete labels is to automatically and accurately fill the missing labels such that a high quality multi-label prediction model can be trained with the completed labels. Comparing to the tremendous amount of work on standard multi-label learning, there are relatively fewer works recently developed for multi-label learning with incomplete labels [2, 3, 4, 15, 20, 23, 28]. These works nevertheless are still limited in exploring the potential complex label dependence information in the label space. None of them have considered incorporating auxiliary label co-occurrence information into the learning process to improve the quality of completed labels and prediction models.

In this paper, we develop a conditional restricted

Appearing in Proceedings of the 18^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2015, San Diego, CA, USA. JMLR: W&CP volume 38.

Boltzmann machine (CRBM) model to address multi-label learning with incomplete labels. Different from the typical probabilistic graphical models, e.g., conditional random fields, used for multi-label learning, which only consider explicit and pre-fixed low-order label dependence relationships for tractable inference, we build a latent layer above the label layer and form a restricted Boltzmann machine model in the output space conditioning on the observed input features. Restricted Boltzmann machines (RBMs) have been shown to be effective in learning high-level features and capturing high-order correlations of the observed variables. We hence expect the CRBM model can effectively encode the high-order label dependence relationships to facilitate label recovery and multi-label prediction model learning. In particular, we formulate the label completion and multi-label prediction model learning as a joint optimization problem, which maximizes the regularized marginal conditional likelihood of the label vectors given the input features under the CRBM model. Moreover, label co-occurrence statistics can be estimated from a large text corpus such as *Wikipedia*. We further extend our model to incorporate such auxiliary label relatedness information as prior knowledge. We develop a Viterbi style EM algorithm to solve the optimization problem produced, which alternatingly trains the CRBM model and recovers the missing labels. To evaluate the proposed model, we compare it with a number of related state-of-the-art methods on four real world multi-label data sets. The experimental results demonstrate the effectiveness of the proposed model on addressing multi-label learning with incomplete label assignments.

2 Related Work and Preliminaries

2.1 Related Work

In multi-label learning problems, each instance can be associated with multiple labels simultaneously. Typically the labels are not independent of each other, but rather demonstrate strong label correlation or dependence patterns. Many multi-label learning works developed in the literature focused on exploiting the label correlation or dependence information in a tractable manner to improve the quality of multi-label prediction, including a few probabilistic graphical model based methods [9, 10, 29]. Recently, there are also a set of label space transformation works developed for multi-label learning [1, 5, 24]. These works induce alternative label representations and perform multi-label learning in the new and typically dimension reduced label space. But none of these works address learning with incomplete labels.

There are a limited number of works that tackled the problem of multi-label learning with incomplete labels [2, 3, 4, 15, 20, 23, 28]. [2] develops a probabilistic model that exploits multi-label correlations and handles missing labels. [3] formulates multi-label classification as a bipartite ranking problem and exploits the group lasso technique to handle incomplete label assignments. [4] presents a fast tagging method which learns two linear mapping matrices from both the input space and the original label space to recover the missing labels. [15] applies stochastic gradient descent to infer missing labels and then trains a stacked model for the final prediction. [20] exploits the information pertaining to partially annotated or unannotated images to achieve semi-supervised learning under a hierarchical Dirichlet process structure. [23] proposes to infer missing labels under transductive settings. [28] presents a generic empirical risk minimization framework for large-scale multi-label learning and accommodates it to missing labels by only training prediction models on known labels and ignoring unknown ones. Different from these methods, our proposed approach is an inductive learning approach. It addresses multi-label learning with incomplete labels by using a latent layer in the output space to capture high-order label dependence relationships for label imputation while simultaneously permitting label prediction in the original label space. Moreover, the previous methods do not exploit free auxiliary resources to infer label relatedness information, while our approach can incorporate such information as model priors.

2.2 Restricted Boltzmann Machines

Restricted Boltzmann machines (RBMs) [21] are special form of undirected graphical models that use hidden variables to model high-order and non-linear regularities of the data. In particular, a RBM is a two-layer bipartite graph with two types of units, the visible units $\mathbf{v} = [v_1, \cdots, v_I]^\top$ and hidden units $\mathbf{h} = [h_1, \cdots, h_J]^\top$. The visible units in one layer correspond to the components of an observation, while the hidden units in the other layer model dependencies between the components of observations. The restriction is that there is no connection between units in the same layer. The RBM represents probability distributions over the random variables under an energy-based model. For an energy function that captures the restricted unit interaction patterns, $E(\mathbf{v}, \mathbf{h}) = -\mathbf{v}^\top W \mathbf{h} - \mathbf{b}^\top \mathbf{v} - \mathbf{c}^\top \mathbf{h}$, the joint probability distribution over $(\mathbf{v}, \mathbf{h})$ can be easily expressed as $P(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} \exp(-E(\mathbf{v}, \mathbf{h}))$, where Z is the normalization factor.

After learning, with the learned model parameters, a RBM can provide a closed-form representation for the distribution underlying the observations. The probability for any subsets of variables can be easily

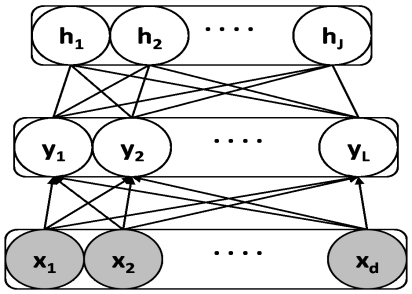

Figure 1: A conditional RBM model.

obtained through conditioning and marginalization. Hence for a partial observation, given the observed visible units, one can sample the remaining visible units to complete the observation [14]. Recently, RBMs have received a lot of attention in deep learning networks; in particular, deep belief networks can be formed by stacking multiple RBMs [12]. Nevertheless, RBMs have been mostly used in learning feature representations in the input feature space. Its potential in modeling the regularities in the output label space has only received limited attention [13, 16, 17] and not yet been well explored to address different learning problems.

3 Proposed Approach

In this section, we present a probabilistic graphical model that uses a restricted Boltzmann machine model to capture high-order label dependence relationships in the output label space and perform effective multi-label learning with incomplete labels. It has the convenient capacity of incorporating auxiliary label relatedness information into the learning framework.

3.1 A Conditional RBM Model

Given training data with incomplete labels for multi-label classification, $D = \{(\mathbf{x}^i, \mathbf{z}^i)\}_{i=1}^N$, where $\mathbf{x}^i \in \mathbb{R}^d$ is the input feature vector for the i-th instance and $\mathbf{z}^i \in \{0,1\}^L$ is the corresponding label indicator vector. We assume that in the label indicator vector $\mathbf{z}^i$, an entry value 1 indicates the existence of the corresponding label, while an entry value 0 indicates an unknown status with a possible missing label. In this work, we use Ω to denote the index set of the observed labels in the training data, such as $(i,j) \in \Omega$ if and only if $\mathbf{z}^i_j = 1$. The existence of missing labels can greatly exacerbate the support sparsity of the labels in the training data and increase the difficulty of learning accurate multi-label prediction models.

To tackle this problem, we propose a conditional restricted Boltzmann machine (CRBM) model for multi-label classification, which has a label layer $\mathbf{y}$ to represent the underlying true label vectors and adds another latent layer $\mathbf{h}$ above the label layer to form a restricted Boltzmann machine. In the literature, RBMs have been effectively used to capture high-order regularities in the input feature space. We expect a conditional RBM in the output label space can effectively capture high-order label dependence information to automatically recover the underlying true label matrix $[\mathbf{y}^1, \cdots, \mathbf{y}^N]$ and learn high-quality multi-label prediction models. The CRBM model is illustrated in Figure 1, which has three layers: the input feature layer $\mathbf{x}$, the output label layer $\mathbf{y}$ and the latent layer $\mathbf{h}$; conditioning on $\mathbf{x}$, the two layers $\mathbf{y}$ and $\mathbf{h}$ form a standard RBM. This CRBM defines a conditional joint distribution over $(\mathbf{y}, \mathbf{h})$, $P(\mathbf{y}, \mathbf{h}|\mathbf{x})$. The conditional marginal distribution of the label vector $\mathbf{y}$ given the observed input feature vector $\mathbf{x}$ is defined as

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \sum_{\mathbf{h}} \exp\left(-E_{cond}(\mathbf{y}, \mathbf{x}) - E_{rbm}(\mathbf{y}, \mathbf{h})\right) \tag{1}$$

where $Z(\mathbf{x})$ is the normalization factor, and $E_{cond}(\mathbf{y}, \mathbf{x})$ and $E_{rbm}(\mathbf{y}, \mathbf{h})$ are two energy functions such as

$$Z(\mathbf{x}) = \sum_{\mathbf{y}} \sum_{\mathbf{h}} \exp\left(-E_{cond}(\mathbf{y}, \mathbf{x}) - E_{rbm}(\mathbf{y}, \mathbf{h})\right) \tag{2}$$

$$E_{cond}(\mathbf{y}, \mathbf{x}) = -\mathbf{y}^\top W \mathbf{x} \tag{3}$$

$$E_{rbm}(\mathbf{y}, \mathbf{h}) = -\mathbf{y}^\top G \mathbf{h} - \mathbf{y}^\top \mathbf{b} - \mathbf{c}^\top \mathbf{h}. \tag{4}$$

The energy function $E_{cond}(\mathbf{y}, \mathbf{x})$ captures the conditional predictive interactions between the input features and the output labels, while the energy function $E_{rbm}(\mathbf{y}, \mathbf{h})$ captures standard RBM interactions between the label layer and the latent layer. Assume there are J latent units in the h layer, we have $\mathbf{h} \in \{0,1\}^J$, while $\mathbf{y} \in \{0,1\}^L$. Then the CRBM model involves the following set of model parameters, $\Theta = \{W \in \mathbb{R}^{L \times d}, G \in \mathbb{R}^{L \times J}, \mathbf{b} \in \mathbb{R}^{L \times 1}, \mathbf{c} \in \mathbb{R}^{J \times 1}\}$. Note under this CRBM model, we still have the conditional independence properties of the standard RBM models such as

$$P(\mathbf{h}|\mathbf{y}, \mathbf{x}) = \prod_{j=1}^{J} P(\mathbf{h}_j|\mathbf{y}, \mathbf{x}),\ P(\mathbf{y}|\mathbf{h}, \mathbf{x}) = \prod_{\ell=1}^{L} P(\mathbf{y}_\ell|\mathbf{h}, \mathbf{x}),$$

and the local conditional probabilities can be easily computed as

$$P(\mathbf{h}_j = 1|\mathbf{y}, \mathbf{x}) = \sigma(\mathbf{y}^\top G_{:j} + \mathbf{c}_j), \tag{5}$$

$$P(\mathbf{y}_\ell = 1|\mathbf{h}, \mathbf{x}) = \sigma(G_{\ell:}\mathbf{h} + \mathbf{b}_\ell + W_{\ell:}\mathbf{x}), \tag{6}$$

where the $\sigma(x)$ denotes the standard sigmoid function, $\sigma(x) = 1/(1 + \exp(-x))$.

Given the training data D with missing labels, we will simultaneously learn the true label vectors and train the CRBM model by maximizing the following regularized conditional marginal likelihood of the labels given the observed input data

$$\max_{\Theta} \max_{\{\mathbf{y}^i\}} \quad \sum_i \log P(\mathbf{y}^i|\mathbf{x}^i) - \frac{\gamma_w}{2}\|W\|_F^2 - \frac{\gamma_g}{2}\|G\|_F^2 - \mu \sum_i \|\mathbf{y}^i\|_1 \quad (7)$$
$$\text{subject to} \quad \mathbf{y}^i \in \{0,1\}^L,\ \mathbf{y}^i_j = 1,\ \forall (i,j) \in \Omega,\ \forall i$$

where γ_w, γ_g, μ are trade-off parameters, $\|\mathbf{y}^i\|_1$ denotes the L1-norm regularizer, and $\|\cdot\|_F$ denotes the Frobenius norm. It is easy to note that the label vectors should be sparse since each instance is typically only assigned very few positive labels from the overall label set. We hence use the L1-norm regularizer to promote label sparsity for the recovered label vectors. The constraints enforce the label vectors to take indicator binary values that are consistent with the label observations in the training data.

3.2 Incorporating Auxiliary Label Relatedness Information

The multiple labels of the classification tasks typically have semantic meanings that can expose some relatedness information, in particular co-occurrence information, between the label concepts. To exploit such relatedness information, we construct a label correlation matrix $\Sigma \in \mathbb{R}^{L\times L}$ using the knowledge extracted from free auxiliary resources such as Wikipedia. In Wikipedia, each article is related to one topic and hence we can express our label concept in terms of the Wikipedia topics as discussed in [19]. Specifically, we collect M topics from Wikipedia and represent each label as a M-dim vector with each entry recording the statistical occurrence information (e.g., occurrence counts, tf-idf feature values) of the *label phrase* in the articles with the corresponding topic. In our experiments, we used the explicit semantic analysis procedure in [8], which uses $M = 389,202$ Wikipedia articles. Then we compute the label correlation matrix Σ by setting its entry Σ_{ij} as the cosine similarity between the M-dim vectors of the two label concepts, c_i and c_j, which naturally captures the co-occurrence information of the labels in the Wikipedia data. Given this label correlation matrix, we can compute a prior distribution $P_0(\mathbf{y})$ over the label vector $\mathbf{y}$ such as

$$P_0(\mathbf{y}) \ \propto\ \exp(\mathbf{y}^\top \Sigma \mathbf{y}) \quad (8)$$

which encodes the prior probabilities of possible label vector configurations based on the auxiliary label correlation knowledge.

We incorporate this auxiliary label correlation matrix into our probabilistic learning model as a regularization term for each label vector $\mathbf{y}^i$. This leads to the following optimization problem

$$\max_{\Theta} \max_{\{\mathbf{y}^i\}} \quad \sum_i \log P(\mathbf{y}^i|\mathbf{x}^i) + \beta \sum_i \log P_0(\mathbf{y}^i) - \frac{\gamma_w}{2}\|W\|_F^2 - \frac{\gamma_g}{2}\|G\|_F^2 - \mu \sum_i \|\mathbf{y}^i\|_1 \quad (9)$$
$$\text{subject to} \quad \mathbf{y}^i \in \{0,1\}^L,\ \mathbf{y}^i_j = 1,\ \forall (i,j) \in \Omega,\ \forall i$$

3.3 Learning Algorithm

The learning problem (9) formulated above has two sets of variables, the set of model parameters Θ and the latent label vector variables $\{\mathbf{y}^i\}$, and it is not a joint convex optimization problem. Let $\mathcal{L}(\Theta, \{\mathbf{y}^i\})$ denote the objective function of (9). We propose to perform learning using a Viterbi style expectation-maximization (EM) algorithm, which alternatingly maximizes the objective function $\mathcal{L}$ with respect to one set of variables given the other set fixed. Specifically, there are two steps: the *model parameter learning step (maximization step)*, where we perform optimization with respect to the set of model parameters Θ given the current label vectors $\{\mathbf{y}^i\}$; and the *label recovery step (Viterbi expectation step)*, where given the current model parameters, we perform optimization to recover the latent label vectors $\{\mathbf{y}^i\}$. We present these two steps below.

3.3.1 Model Parameter Learning

Given the current recovery of the missing labels in each label vector $\mathbf{y}^i$, the CRBM model is equivalent to an extended standard RBM model; we can see the joint energy function $E(\mathbf{y}, \mathbf{h}, \mathbf{x}) = E_{cond}(\mathbf{y}, \mathbf{x}) + E_{rbm}(\mathbf{y}, \mathbf{x})$ in Eq.(1) only extends the standard RBM model parameters by adding a $W\mathbf{x}$ term to the original $\mathbf{b}$ parameter vector. A standard gradient ascent algorithm for learning undirected graphical models can iteratively update the set of model parameters Θ with $\theta = \theta + \epsilon\Delta\theta$ for each $\theta \in \Theta$, where ϵ and $\Delta\theta$ are the learning rate and direction of the update respectively. In the standard gradient ascent, the update direction $\Delta\theta$ is simply the partial gradient of the objective function with respect to the model parameter θ:

$$\Delta\theta = \frac{\partial \mathcal{L}}{\partial \theta} = -\sum_i \begin{pmatrix} \sum_{\mathbf{h}} P(\mathbf{h}|\mathbf{y}^i, \mathbf{x}^i) \frac{\partial E(\mathbf{y}^i, \mathbf{h}, \mathbf{x}^i)}{\partial \theta} \\ -\sum_{\mathbf{h}} \sum_{\mathbf{y}} P(\mathbf{y}, \mathbf{h}|\mathbf{x}^i) \frac{\partial E(\mathbf{y}, \mathbf{h}, \mathbf{x}^i)}{\partial \theta} \end{pmatrix} - \theta \left(\gamma_g I_{[\theta \in G]} + \gamma_w I_{[\theta \in W]}\right) \quad (10)$$

where $I_{[\cdot]}$ denotes an indicator function which takes value 1 when the given condition in the brackets is true. However, the second term within the first set

of brackets in (10) involves exponential complexity of summation. To avoid this computational difficulty, we hence use the standard k-step contrastive divergence (CD-k) algorithm [11] to perform model parameter learning in this extended RBM model. The CD-k algorithm is a stochastic approximate gradient ascent algorithm. For each i-th instance, it runs the MCMC chain for k steps, starting from the pre-given $\mathbf{y}^{(0)_i} = \mathbf{y}^i$ vector. In each r-th step, it samples $\{\mathbf{h}_j^{(r)_i}\}$ given the previous $\mathbf{y}^{(r-1)_i}$ and then samples $\{\mathbf{y}_\ell^{(r)_i}\}$ given $\mathbf{h}^{(r)_i}$ according to the individual binomial conditional distributions of $\mathbf{h}_j$ and $\mathbf{y}_\ell$ given in Eq.(5) and Eq.(6) respectively. Finally it approximates the gradient ascent direction with the samples obtained at the k-th step, such that

$$\Delta\theta = -\sum_i \begin{pmatrix} \sum_{\mathbf{h}} P(\mathbf{h}|\mathbf{y}^i, \mathbf{x}^i) \frac{\partial E(\mathbf{y}^i, \mathbf{h}, \mathbf{x}^i)}{\partial \theta} \\ -\sum_{\mathbf{h}} P(\mathbf{h}|\mathbf{y}^{(k)_i}, \mathbf{x}^i) \frac{\partial E(\mathbf{y}^{(k)_i}, \mathbf{h}, \mathbf{x}^i)}{\partial \theta} \end{pmatrix} - \theta \left(\gamma_g I_{[\theta \in G]} + \gamma_w I_{[\theta \in W]}\right) \quad (11)$$

In our experiments, we used CD-1 algorithm with $k = 1$, which has been shown to work well in previous studies on standard RBMs.

3.3.2 Latent Label Recovery

Given the current CRBM model parameters Θ, the optimization problem in (9) can be decomposed into a set of N independent sub-optimization problems, one for each instance label vector $\mathbf{y}^i$:

$$\mathbf{y}^i = \arg\max_{\mathbf{y}} \quad \log P(\mathbf{y}|\mathbf{x}^i) + \beta \mathbf{y}^\top \Sigma \mathbf{y} - \mu \|\mathbf{y}\|_1 \quad (12)$$
$$\text{subject to} \quad \mathbf{y} \in \{0,1\}^L,\ \mathbf{y}_j = 1,\ \forall (i,j) \in \Omega.$$

Let $\mathcal{L}(i, \mathbf{y})$ denote the objective function in (12). Note the normalization factor $Z(\mathbf{x}^i)$ (see its definition in Eq.(2)) for computing $P(\mathbf{y}|\mathbf{x}^i)$ is independent of the $\mathbf{y}$ variables, and hence the objective function $\mathcal{L}(i, \mathbf{y})$ can be simplified into

$$\mathcal{L}(i, \mathbf{y}) = \sum_{j=1}^J \log\left(1 + \exp(\mathbf{y}^\top G_{:j} + \mathbf{c}_j)\right) + \mathbf{y}^\top \mathbf{b} + \mathbf{y}^\top W \mathbf{x}^i + \beta \mathbf{y}^\top \Sigma \mathbf{y} - \mu \|\mathbf{y}\|_1 \quad (13)$$

Moreover, given the constraint that $\mathbf{y}$ has nonnegative values, the $L1$ norm regularizer $\|\mathbf{y}\|_1$ is equivalent to the sum of the vector entries and the objective function is a smooth function. We then relax the integer constraint $\mathbf{y} \in \{0,1\}^L$ into $0 \le \mathbf{y} \le 1$, and use a projected gradient ascent algorithm with backtracking line search to perform relaxed optimization.

In each iteration t of the projected gradient ascent algorithm, given the current point $\mathbf{y}^{(t)}$, the next point $\mathbf{y}^{(t+1)}$ can be reached by

$$\mathbf{y}^{(t+1)} = \text{Proj}\left(\mathbf{y}^{(t)} + \eta^* \nabla_{\mathbf{y}^{(t)}} \mathcal{L}(i, \mathbf{y})\right), \quad (14)$$

where $\nabla_{\mathbf{y}^{(t)}} \mathcal{L}(i, \mathbf{y})$ is the gradient vector value at the current point, η^* is the optimal step size found by backtracking line search that maximizes the objective function. The projection operator, $Proj(\cdot)$, projects the input vector into the feasible region defined by the constraints, and its j-th entry is defined as

$$\text{Proj}\left(\widehat{\mathbf{y}}_j\right) = \begin{cases} 1 & \text{if } (i,j) \in \Omega \\ \max(0, \min(1, \widehat{\mathbf{y}}_j)) & \text{otherwise} \end{cases} \quad (15)$$

After converging to a local optimal solution $\mathbf{y}^*$, we can round it back to $\{0,1\}$ values to recover the label vector $\mathbf{y}^i$. In this procedure, the label vector is recovered by integrating both the predictive information from the input features, the high-order label dependence information captured in the CRBM model, and the auxiliary label relatedness information.

Testing Phase In the test phase, given an instance $\mathbf{x}$, and the CRBM model learned in the training process, we first initialize $\mathbf{y} = \sigma(W\mathbf{x})$ by only considering the input feature information. Then we infer the $\mathbf{y}$ labels by using the latent label recovery step above with an empty Ω set.

4 Experiment and Results

To evaluate the proposed conditional restricted Boltzmann machine (*CRBM*) model for multi-label learning with incomplete labels, we conducted experiments on four diverse types of real-world multi-label data sets: *Corel5K, Mediamill, CLEF2010* and *Delicious*. *Corel5K* [7] is an image data set, which contains 5000 instances and 374 labels, with an average of 3.5 labels assigned to each instance. *Mediamill* [22] is a video retrieval data set with 43,907 instances and 101 labels. On average, each instance in this data set has 4.4 labels. *CLEF2010* [25] contains 10,000 images and 93 labels, with an average of 11.7 labels for each image. We rescaled each image to 256×256 and then extracted 512-dimension GIST [18] features to use. *Delicious* [26] is a text data set, which contains 16,105 instances and 983 labels, with an average of 19.0 labels assigned to each instance.

We compared the proposed *CRBM* approach with the following state-of-the-art multi-label learning methods that are tailored for incomplete label assignment scenarios: (1) A multi-label classification method with label correlations and missing labels (*LCML*) [2]. (2) A multi-label ranking with group lasso (*MLRGL*) algorithm [3]. (3) A fast tagging method (*FastTag*) [4]. The proposed *CRBM* model has the capacity of incorporating auxiliary label relatedness information. In the experiments, we exploited the auxiliary label information from Wikipedia by computing label concepts with the explicit semantic analysis method [8], as described

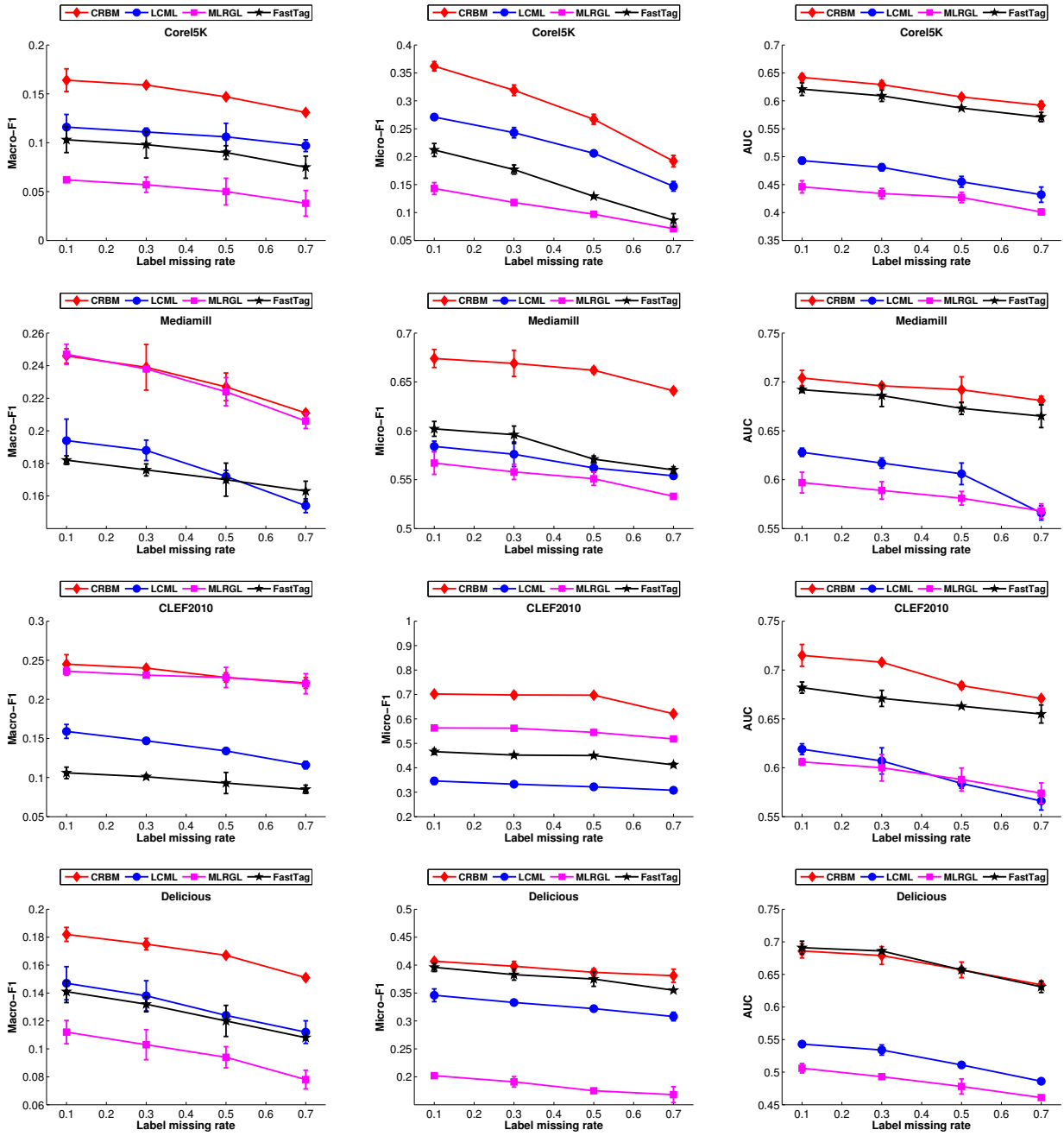

Figure 2: Classification results on the four data sets in terms of the three evaluation measures: Each row presents the results on each data set; each column presents the results in terms of each of the three measures.

in Section 3.2.

For each data set, we randomly selected 3,000 instances as training data while keeping the rest as test data. We randomly selected a fraction, σ, of the observed labels on the training data to drop and simulate the missing labels. To perform parameter selection for each experimented approach, we further split the training data into a training set with 2,400 instances and a validation set with 600 instances. On the training set, we randomly dropped 10% of the labels to simulate the missing label scenario for the parameter selection process; we train each method on the training set and evaluate its prediction performance on the validation set for parameter selection. For the proposed approach, we fixed $\gamma_g = \gamma_w = 0.01$ and performed parameter selection on μ and β with μ chosen from $\{0.01, 0.05, 0.1, 0.5, 1\}$ and β chosen from $\{0.01, 0.05, 0.1, 0.5, 1, 5\}$. The number of hidden units, J, was set roughly proportional to the original label di-

mension of each data set: We used J=50 for Corel5K, J=20 for Mediamill and CLEF2010, and J=100 for Delicious. For the other comparison methods, we performed parameter selection for their parameters from the ranges of values suggested in their original papers. We measured the prediction performance of all the methods using three standard multi-label evaluation metrics: Macro-F1, Micro-F1 and the AUC (area under the curve) criteria. All the results reported in this section are averages on *five* repeated runs with different random data partitions.

4.1 Classification Results on the Test Data

To investigate the learning capacity of the comparison methods in the scenario of incomplete labels, we conducted experiments with a range of different training label missing rates, $\sigma \in \{10\%, 30\%, 50\%, 70\%\}$. For each given training label missing rate, the average test performance over five runs were recorded on each data set for each comparison method. The comparison results in terms of the three evaluation measures on the four data sets are reported in Figure 2.

We can see that among the three comparison methods, *LCML, MLRGL* and *FastTag*, each of them exhibits strength on different data sets and with different evaluation measures. *LCML* outperforms both *MLRGL* and *FastTag* on *Corel5K* in terms of Macro-F1 and Micro-F1 and on *Delicious* in terms of Macro-F1, across the range of different training label missing rates, but produces the worst results on *CLEF2010* in terms of Micro-F1. *MLRGL* produces the best results among the three methods on *CLEF2010* in terms of Macro-F1 and Micro-F1 and on *Mediamill* in terms of Macro-F1, across the range of different training label missing rates, but produces the worst results on *Corel5K* and *Delicious* across all three measures and on *Mediamill* in terms of Micro-F1 and AUC. *FastTag* outperforms *LCML* and *MLRGL* across all the four data sets in terms of AUC measure and on *Mediamill* and *Delicious* in terms of Micro-F1, but produces the worst results on *CLEF2010* and *Mediamill* in terms of Macro-F1. On the other hand, the proposed approach *CRBM* consistently produces the best results and outperforms all the three comparison methods across different label missing rates on all the four data sets in terms of almost all the three evaluation measures, except on *Mediamill* with $\sigma = 10\%$ in terms of Macro-F1 and on *Delicious* with $\sigma \in \{10\%, 30\%\}$ in terms of AUC. Even in the three cases where *CRBM* fails to produce the best results, it produces the second best results that are very close to the best ones. These results demonstrate the efficacy of the proposed approach on handling multi-label learning with missing labels in different scenarios. We also noted in our experiments that the proposed *CRBM* has comparable training time with the other comparison methods.

4.2 Label Recovery on the Training Data

For all the comparison methods used in our experiments, we have also investigated their capacity of recovering missing labels on the training data. Specifically, for each method, after training a multi-label prediction model on the training data that have incomplete label assignments, we apply the prediction model on each training instance to obtain a predicted label vector, which is expected to recover the missing labels in the original label vector. We then evaluate the performance of missing label recovery using a *missing label recovery accuracy* measure, which is defined as the ratio between the number of correctly recovered missing labels and the number of total missing labels on the training data.

The average missing label recovery results for all the comparison methods on the four data sets are reported in Table 1.[1] We can see that with the increase of the label missing rate, the performance of all methods naturally degrades. But in general all the methods produce reasonably good results, even with $\sigma = 70\%$. *FastTag* demonstrates a good label recovery capacity by consistently outperforming *LCML* and *MLRGL*. But our proposed method, *CRBM*, consistently outperforms *FastTag* and produces the best results across all the data sets and different label missing rates.

4.3 Impact of Auxiliary Knowledge

We have also further studied the impact of different auxiliary knowledge within our proposed model on the image data set *Corel5K*. In addition to exploiting Wikipedia data based on the explicit semantic analysis (ESA) technique, which we denote as *CRBM-Wiki-ESA* here, we also tested two alternative methods of extracting auxiliary knowledge from Wikipedia and an image resource by learning word embedding and co-occurrence statistics respectively. The word embedding (WE) method exploits the Wikipedia resource as well. But different from the explicit semantic analysis method, we use the neural network word embedding technique in [6] to learn word embedding vectors from Wikipedia articles, which provides semantic vector representations for all the labels we have for *Corel5K*. In particular, we used 50-dimension vectors for the labels. The label correlation matrix Σ was computed based on the cosine similarity between the label vectors. We denote this method as *CRBM-*

[1] Due to space limitation, we only reported the results for missing rates $\sigma \in \{30\%, 70\%\}$. The comparison results with $\sigma \in \{10\%, 50\%\}$ are quite similar to the reported ones.

Table 1: Missing label recovery accuracy results (mean±std) on the four data sets.

Label Missing Rate	Methods	Corel5K	Mediamill	CLEF2010	Delicious
30%	CRBM	**0.941 ± 0.013**	**0.902 ± 0.013**	**0.904 ± 0.001**	**0.936 ± 0.011**
	LCML	0.897 ± 0.010	0.839 ± 0.005	0.807 ± 0.013	0.903 ± 0.000
	MLRGL	0.874 ± 0.007	0.866 ± 0.010	0.828 ± 0.013	0.852 ± 0.006
	FastTag	0.926 ± 0.008	0.900 ± 0.003	0.898 ± 0.011	0.914 ± 0.002
70%	CRBM	**0.885 ± 0.007**	**0.847 ± 0.013**	**0.855 ± 0.002**	**0.886 ± 0.001**
	LCML	0.804 ± 0.003	0.754 ± 0.009	0.731 ± 0.010	0.842 ± 0.009
	MLRGL	0.807 ± 0.007	0.819 ± 0.009	0.767 ± 0.002	0.804 ± 0.001
	FastTag	0.877 ± 0.009	0.845 ± 0.012	0.847 ± 0.009	0.853 ± 0.001

Table 2: The classification results (mean±std) on *Corel5K* with different auxiliary knowledge.

Measure	Label Missing Rate	CRBM-Wiki-ESA	CRBM-Wiki-WE	CRBM-SUN-CS	CRBM+ø
Macro-F1	30%	**0.159 ± 0.001**	0.138 ± 0.002	0.156 ± 0.002	0.125 ± 0.005
	70%	**0.131 ± 0.002**	0.113 ± 0.005	0.127 ± 0.007	0.097 ± 0.005
Micro-F1	30%	**0.319 ± 0.001**	0.288 ± 0.012	0.307 ± 0.002	0.267 ± 0.011
	70%	**0.192 ± 0.002**	0.165 ± 0.005	0.187 ± 0.006	0.153 ± 0.002
AUC	30%	**0.629 ± 0.001**	0.598 ± 0.008	0.617 ± 0.002	0.577 ± 0.002
	70%	**0.592 ± 0.002**	0.505 ± 0.007	0.581 ± 0.011	0.496 ± 0.008

Wiki-WE. The co-occurrence statistics (CS) method is used to extract knowledge from a more relevant auxiliary resource, i.e., a large scale image data set *SUN* [27], which contains 908 scene categories and 3,819 object categories. We calculated the label correlation matrix Σ for labels in *Corel5K* based on the label co-occurrence information presented in *SUN*, such as $\Sigma_{ij} = n_{ij}/(n_i + n_j)$, where n_i and n_j denote the numbers of occurrences of the i-th and j-th label concepts of *Corel5K* in *SUN*, and n_{ij} denotes the number of co-occurrences of the two label concepts in *SUN*. We denote this method as *CRBM-SUN-CS*. We compared these three variants of the proposed model with a baseline *CRBM-ø* model that ignores the auxiliary label information. The results with two different label missing rates are reported in Table 2.

We can see that the three variants of *CRBM* that exploit auxiliary knowledge all consistently outperform the baseline model. This suggests that the label relatedness knowledge extracted from free auxiliary resources is in general helpful for handling missing training labels and our proposed model provides the proper capacity of exploiting such knowledge. Among the three variants, the improvements achieved by *CRBM-Wiki-WE* over the baseline *CRBM-ø* are much smaller than the other two variant methods. This is reasonable since the similarity between the label embedding vectors extracted by *CRBM-Wiki-WE* in some cases may not reflect the label co-occurrence information. *CRBM-Wiki-ESA* and *CRBM-SUN-CS* outperform *CRBM-ø* with large margins across all the evaluations as they both incorporate label co-occurrence information. Moreover *CRBM-Wiki-ESA* produces the best results among all the methods across all the learning scenarios and evaluation measures. These results again justified our proposed model in incorporating auxiliary label correlation knowledge from Wikipedia.

5 Conclusion

In this paper, we proposed a novel conditional restricted Boltzmann machine (CRBM) model to capture high-order label dependence relationships and facilitate multi-label learning with incomplete labels. This model also incorporates label correlation information extracted from auxiliary resources as prior regularization knowledge. Under this model, we formulated the label completion and multi-label prediction model learning as a joint optimization problem, which maximizes the regularized marginal conditional likelihood of the label vectors given the input features. We developed a Viterbi style EM algorithm to solve the joint optimization problem produced. Experiments were conducted over four real world multi-label data sets to compare the proposed approach with a number of state-of-the-art methods. The experimental results demonstrated the efficacy of the proposed model on addressing multi-label learning with incomplete labels.

Acknowledgements

This research was supported in part by NSF grant IIS-1422127.

References

[1] W. Bi and J. Kwok. Efficient multi-label classification with many labels. In *Proc. of ICML*, 2013.

[2] W. Bi and J. Kwok. Multilabel classification with label correlations and missing labels. In *Proceedings of AAAI*, 2014.

[3] S. Bucak, R. Jin, and A. Jain. Multi-label learning with incomplete class assignments. In *Proceedings of CVPR*, 2011.

[4] M. Chen, A. Zheng, and K. Weinberger. Fast image tagging. In *Proceedings of ICML*, 2013.

[5] Y. Chen and H. Lin. Feature-aware label space dimension reduction for multi-label classification. In *Proceedings of NIPS*, 2012.

[6] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12:2493–2537, 2011.

[7] P. Duygulu, K. Barnard, J. Freitas, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Proceedings of ECCV*, 2002.

[8] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of IJCAI*, 2007.

[9] N. Ghamrawi and A. McCallum. Collective multi-label classification. In *Proceedings of CIKM*, 2005.

[10] Y. Guo and S. Gu. Multi-label classification using conditional dependency networks. In *Proceedings of IJCAI*, 2011.

[11] G. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14:1771–1800, 2002.

[12] G. Hinton, S. Osindero, and Y. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554, 2006.

[13] A. Kae, K. Sohn, H. Lee, and E. Learned-Miller. Augmenting CRFs with Boltzmann machine shape priors for image labeling. In *Proceedings of CVPR*, 2013.

[14] J. Kivinen and C. Williams. Multiple texture Boltzmann machines. In *Proceedings of AISTATS*, 2012.

[15] X. Kong, Z. Wu, J. Li, R. Zhang, and P. Yu. Large-scale multi-label learning with incomplete label assignments. In *Proceedings of SDM*, 2014.

[16] H. Larochelle and Y. Bengio. Classification using discriminative restricted Boltzmann machines. In *Proceedings of ICML*, 2008.

[17] V. Mnih, H. Larochelle, and G. Hinton. Conditional restricted Boltzmann machines for structured output prediction. In *Proc. of UAI*, 2011.

[18] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001.

[19] A. Panchenko. A study of heterogeneous similarity measures for semantic relation extraction. In *Proceedings of the Joint Conference JEP-TALN-RECITAL*, 2012.

[20] Z. Qi, M. Yang, Z. Zhang, and Z. Zhang. Mining partially annotated images. In *Proceedings of KDD*, 2011.

[21] P. Smolensky. Information processing in dynamical systems: Foundations of harmony theory. *Parallel distri. processing: Explorations in the microstructure of cognition*, 1:194–281, 1986.

[22] C. Snoek, M. Worring, J. van Gemert, J. Geusebroek, and A. Smeulders. The challenge problem for automated detection of 101 semantic concepts in multimedia. In *Proceedings of MM*, 2006.

[23] Y. Sun, Y. Zhang, and Z. Zhou. Multi-label learning with weak label. In *Proc. of AAAI*, 2010.

[24] F. Tai and H. Lin. Multilabel classification with principal label space transformation. *Neural Comput.*, 24(9):2508–2542, Sept. 2012.

[25] T. Tsikrika and J. Kludas. The Wikipedia image retrieval task. In *ImageCLEF*, volume 32 of *The Information Retrieval Series*. Springer, 2010.

[26] G. Tsoumakas, I. Katakis, and I. Vlahavas. Effective and efficient multilabel classification in domains with large number of labels. In *Proceedings ECML/PKDD Workshop on Mining Multidimensional Data*, 2008.

[27] J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Proceedings of CVPR*, 2010.

[28] H. Yu, P. Jain, P. Kar, and I. Dhillon. Large-scale multi-label learning with missing labels. In *Proceedings of ICML*, 2014.

[29] J. Zaragoza, L. Sucar, E. Morales, C. Bielza, and P. Larranaga. Bayesian chain classifiers for multidimensional classification. In *Proceedings of IJCAI*, 2011.