Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)

# Learning with Marginalized Corrupted Features and Labels Together

**Yingming Li[†], Ming Yang[‡], Zenglin Xu[†], and Zhongfei (Mark) Zhang[‡]**
[†] School of Computer Science and Engineering, Big Data Research Center
University of Electronic Science and Technology of China
[‡]Department of Computer Science, State University of New York at Binghamton, NY, USA
yingming.li01@gmail.com, myang@binghamton.edu,
zenglin@gmail.com, zhongfei@cs.binghamton.edu

## Abstract

Tagging has become increasingly important in many real-world applications noticeably including web applications, such as web blogs and resource sharing systems. Despite this importance, tagging methods often face difficult challenges such as limited training samples and incomplete labels, which usually lead to degenerated performance on tag prediction. To improve the generalization performance, in this paper, we propose Regularized Marginalized Cross-View learning (RMCV) by jointly modeling on attribute noise and label noise. In more details, the proposed model constructs infinite training examples with attribute noises from known exponential-family distributions and exploits label noise via marginalized denoising autoencoder. Therefore, the model benefits from its robustness and alleviates the problem of tag sparsity. While RMCV is a general method for learning tagging, in the evaluations we focus on the specific application of multi-label text tagging. Extensive evaluations on three benchmark data sets demonstrate that RMCV outstands with a superior performance in comparison with state-of-the-art methods.

## Introduction

Tagging has attracted a great attention in many real-world applications noticeably including web applications such as web blogs and resource sharing systems. It aims at improving the information organization and management by assigning textual descriptions or key-words (called tags) to information object. With typically explosive amount of data to be tagged, it is very time-consuming and labor-intensive for manually labeling data. Consequently, machine learning techniques for tagging have become an effective alternative. In order to reduce the manual effort, many fast tagging methods (Elisseeff and Weston 2001; Yu, Yu, and Tresp 2005; Liu, Jin, and Yang 2006; Zhang and Zhou 2007; Hsu et al. 2009; Zhang and Zhou 2014; Liu and Tsang 2015) have been developed. But most of these studies assume that the amount of training data is sufficient and the given training labels are complete. In contrast, the real-world applications of tagging often face two challenges: limited training examples and incomplete labels.



For the problem of limited training data, a simple approach is to extend the training set explicitly with artificially created examples (Burges and Schölkopf 1996). However, it lacks elegance and the computational cost of processing the extra corrupted examples is prohibitively expensive for most real-world applications. Many algorithms have been developed to consider implicit approaches to tagging objects that are subject to corruptions. (Bishop 1994) shows that training with attribute noise is equivalent to adding a regularization term to the error function. (Globerson and Roweis 2006) introduces a minimax framework under a worst case scenario of feature deletion at test time. (van der Maaten et al. 2013) learns invariant predictors with marginalized corrupted features (MCF), which aims to minimize the expected value of the reconstruction error under the corrupting distribution. MCF enjoys the property of scaling linearly in the number of training samples while ignoring the influence of the incomplete training label set.

On the other hand, recent studies, for example (Chen, Zheng, and Weinberger 2013), consider training with label noise to mitigate the influence of incomplete training label set. It assumes that the given label set is incomplete and proposes a co-regularized learning method to enrich the incomplete user tags. The resulting algorithm significantly improves over the prior state-of-the-art in reducing training and testing time. However, it still suffers from learning with limited training examples, which makes it difficult to develop robust predictors that generalize well to test data.

To improve the generalization performance of tagging, it is necessary to consider both attribute noise and label noise. To achieve this goal, we propose to train robust predictors with attribute noise and label noise simultaneously. In particular, we present the Regularized Marginalized Cross-View learning (referred as RMCV) model, which learns predictors with marginalized corrupted features and labels together. Conceptually, this is equivalent to training the model with infinitely many (corrupted) training examples under attribute noise and label noise together. On the one hand, RMCV extends the general cross-view learning with marginalized corrupted features by corrupting training examples with noise from known distributions within the exponential family; on the other hand, it exploits label noise to enrich the incomplete training label set through a marginalized denoising autoencoder regularization. In addition, it also leads to an

optimization problem which is jointly convex and can be solved through alternative optimization with simple closed-form updates.

While RMCV is a general method, we demonstrate the promise through extensive evaluations in the specific application to multi-label text tagging using real data sets in comparison with the peer methods in the literature.

## Related Work

The noise among the data is catergorized into two types: attribute noise and label noise (Zhu and Wu 2004). On one hand, previous work (Sietsma and Dow 1991) has demonstrated that training with feature noise can lead to improvement in generalization. It has been proved that such training with feature noise is equivalent to a form of regularization in which an additional term is added to the error function (Bishop 1994; Webb 1994). Through modifying the error criterion with a specific regularization term, (Webb 1994) proposes a least square approach, which generalizes to situations in which data samples are corrupted by noise in the input variables. (Burges and Schölkopf 1996) explicitly corrupts the training samples to incorporate invariance with the known transformations.

Further, minimax formulation is proposed to minimize the loss under specific adversarial worst-case feature corruption scenario (Globerson and Roweis 2006). (Xu, Caramanis, and Mannor 2009) considers minimizing the worst possible empirical error under adversarial disturbances on samples. In addition, (Chechik et al. 2008) proposes a max-margin learning framework to classify the incomplete data without any completion of the missing features. (Teo et al. 2007) formulates invariant learning as a convex problem and considers maximum margin loss with invariances.

Recent work proposes dropout training to combat overfitting by artificially corrupting the training data. Dropout training is introduced by (Hinton et al. 2012) as a way to control overfitting by randomly dropping subsets of features at each iteration of a training process. By casting dropout training as regularization, (Wager, Wang, and Liang 2013) present a semi-supervised algorithm, which uses unlabeled data to produce an adaptive regularizer. (Srivastava et al. 2014) shows that dropout improves the performance of supervised neural network learning. In addition, (van der Maaten et al. 2013) proposes to corrupt training examples with noise from known exponential-family distribution.

On the other hand, there are a number of denoising methods for label noise problem. They can be further classified into two categories: filtered preprocessing of the data and robust design of the algorithms. In the former category, filtered preprocessing is developed to remove the noise from the training set as much as possible (Van Hulse and Khoshgoftaar 2006). For the latter category, robust algorithms are designed to reduce the impact of the noise in the classification (Lin and de Wang 2004). (Biggio, Nelson, and Laskov 2011) investigates the robustness of SVMs against adversarial label noise and proposes a strategy to improve the robustness of SVMs based on a kernel matrix correction. In addition, tag refinement is considered as auxiliary work for image noisy tagging in the literature (Wang et al. 2007). Inspired by the recent success of denoising autoencoder (Vincent et al. 2008; Chen et al. 2012), (Chen, Zheng, and Weinberger 2013) proposes to enrich the user tags with marginalized denoising autoencoder.

Among the existing work, the closest work to ours are FastTag (Chen, Zheng, and Weinberger 2013) and MCF (van der Maaten et al. 2013). The former focuses on enriching the user tags with marginalized denoising autoencoder. The latter learns robust predictors by corrupting the training examples with a specific noise distribution. We note that the difference from our work is significant as our method is to combine both merits of the two methods by learning with marginalized corrupted features and marginalized denosing autoencoder regularization together. Consequently, our method outperforms the above two models, which has been demonstrated in the experiments.

## Learning with Marginalized Corrupted Features and Labels Together

In this section, we first develop a novel cross-view learning method, the Marginalized Cross-View learning (MCV). Further, we incorporate the label noise into MCV through a marginalized denoising autoencoder (MDA) regularization. Consequently, RMCV is presented to solve the problem of learning with marginalized corrupted features and labels together.

**Notations.** In this paper, matrices are written in bold uppercase letters and vectors are written in bold lowercase letters. Suppose that we have $n$ samples and each sample has $T$ possible tags. Let the training data be denoted by $\mathcal{D} = \{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_n, \mathbf{y}_n)\} \subset \mathbb{R}^D \times \{0,1\}^T$, where each vector $\mathbf{x}_i \in \mathbb{R}^D$ represents the input features from the $i^{th}$ sample and each $\mathbf{y}_i$ is the corresponding tag vector. Let $\mathbf{W} \in \mathbb{R}^{T \times D}$ denote the linear regression mapping and $\mathbf{B} \in \mathbb{R}^{T \times T}$ represent the tag enrichment mapping.

### Marginalized Cross-View Learning

To derive the framework of marginalized cross-view learning, we start by defining a corrupting distribution that specifies how training examples $\mathbf{x}$ are transformed into corrupted versions $\tilde{\mathbf{x}}$. We assume that the corrupting distribution factorizes over dimensions and that each individual distribution belongs to the natural exponential family. Its particular form is given by:

$$p(\tilde{\mathbf{x}}|\mathbf{x}) = \prod_{d=1}^{D} P_E(\tilde{x}_d|x_d;\eta_d) \tag{1}$$

where $\eta_d$ indicates parameters of the corrupting distribution on dimension $d$.

Based on the produced corrupted examples, we first propose a cross-view regularized learning by interpreting training data (corrupted samples with incomplete tags) as unlabeled multi-view data. In particular, we obtain two subtasks: 1) training a classifier $\tilde{\mathbf{x}}_i \rightarrow \mathbf{W}\tilde{\mathbf{x}}_i$ that predicts the complete tag set from corrupted observations, and 2) training a mapping $\mathbf{y}_i \rightarrow \mathbf{B}\mathbf{y}_i$ to enrich the existing incomplete

Table 1: The probability density function (PDF), mean, and variance of several corrupting distributions. In particular, blankout corruption is the unbiased version of regular feature dropout.

| Distribution | PDF | $\mathbb{E}[\tilde{x}_{id}]_{p(\tilde{x}\|x_{id})}$ | $\mathbf{V}[\tilde{x}_{id}]_{p(\tilde{x}\|x_{id})}$ |
|---|---|---|---|
| Blankout noise | $p(\tilde{x}_{id}=0)=q_d$<br>$p(\tilde{x}_{id}=\frac{1}{1-q_d}x_{id})=1-q_d$ | $x_{id}$ | $\frac{q_d}{1-q_d}x_{id}^2$ |
| Gaussian noise | $p(\tilde{x}_{id}\|x_{id})=\mathcal{N}(\tilde{x}_{id}\|x_{id},\sigma^2)$ | $x_{id}$ | $\sigma^2$ |
| Laplace noise | $p(\tilde{x}_{id}\|x_{id})=Lap(\tilde{x}_{id}\|x_{id},\lambda)$ | $x_{id}$ | $2\lambda^2$ |
| Poisson noise | $p(\tilde{x}_{id}\|x_{id})=Pois(\tilde{x}_{id}\|x_{id})$ | $x_{id}$ | $x_{id}$ |

tag vector $\mathbf{y}_i$ by estimating a mapping function $\mathbf{B}$ which captures the tags' co-occurrence relationships. We train both tasks simultaneously and force a cross-view agreement by minimizing

$$\frac{1}{n}\sum_{i=1}^{n}||\mathbf{B}\mathbf{y}_i-\mathbf{W}\tilde{\mathbf{x}}_i||^2 \tag{2}$$

where $\mathbf{B}\mathbf{y}_i$ is the enriched tag vector for the $i$-th training sample and $\mathbf{W}$ represents the linear classifier which tries to predict the enriched tags based on corrupted samples.

Taking inspiration from the idea of (Burges and Schölkopf 1996) by selecting each sample of the training set $\mathcal{D}=\{(\mathbf{x}_i,\mathbf{y}_i)\}_{i=1}^n$ and corrupting it $m$ times by following the corrupting distribution, we can create corresponding corrupted examples $\tilde{\mathbf{x}}_{ij}$ (with $j=1,\ldots,m$) for each $\mathbf{x}_i$. Further, we construct a new data set $\tilde{\mathcal{D}}$ with $|\tilde{\mathcal{D}}|=mn$. Thus, the above cross-view learning loss function can be rewritten as

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{m}\sum_{j=1}^{m}||\mathbf{B}\mathbf{y}_i-\mathbf{W}\tilde{\mathbf{x}}_{ij}||^2 \tag{3}$$

where $\tilde{\mathbf{x}}_{ij}\sim p(\tilde{\mathbf{x}}_{ij}|\mathbf{x}_i)$.

When $m\to\infty$, we can use the weak law of larger numbers and rewrite the loss function as its expectation (Duda, Hart, and Stork 2001)

$$\begin{aligned}&\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[\|\mathbf{B}\mathbf{y}_i-\mathbf{W}\tilde{\mathbf{x}}_i\|^2]_{p(\tilde{\mathbf{x}}_i|\mathbf{x}_i)}\\ =&\frac{1}{n}\text{trace}\left(\mathbf{W}\left(\sum_{i=1}^{n}\mathbb{E}\left[\tilde{\mathbf{x}}_i\right]\mathbb{E}\left[\tilde{\mathbf{x}}_i\right]^\top+\mathbf{V}\left[\tilde{\mathbf{x}}_i\right]\right)\mathbf{W}^\top\right.\\ &\left.-2\mathbf{B}\sum_{i=1}^{n}\mathbf{y}_i\mathbb{E}\left[\tilde{\mathbf{x}}_i^\top\right]\mathbf{W}^\top+\mathbf{B}\sum_{i=1}^{n}\mathbf{y}_i\mathbf{y}_i^\top\mathbf{B}^\top\right)\end{aligned} \tag{4}$$

where $\mathbf{V}[\mathbf{x}]$ is a diagonal $D\times D$ matrix representing the variance of $\mathbf{x}$, and all the expectations are under $p(\tilde{\mathbf{x}}_i|\mathbf{x}_i)$.

*Marginalized cross-view learning* is referred to minimizing the above expected value of the loss under a specific corrupting distribution.

Define $\mathbf{P}_{xy}=\sum_{i=1}^{n}\mathbf{y}_i\mathbb{E}\left[\tilde{\mathbf{x}}_i^\top\right]$ and $\mathbf{Q}_x=\sum_{i=1}^{n}\mathbb{E}\left[\tilde{\mathbf{x}}_i\right]\mathbb{E}\left[\tilde{\mathbf{x}}_i\right]^\top+\mathbf{V}\left[\tilde{\mathbf{x}}_i\right]$, and then we can rewrite the loss in Eq.(4) as follows:

$$\frac{1}{n}\text{trace}\left(\mathbf{W}\mathbf{Q}_x\mathbf{W}^\top-2\mathbf{B}\mathbf{P}_{xy}\mathbf{W}^\top+\mathbf{B}\mathbf{Y}\mathbf{Y}^\top\mathbf{B}^\top\right) \tag{5}$$

To minimize the expected loss under the particular corruption model, we need to compute the variance of the particular corrupting distribution, which is practical for all the exponential-family distributions. The mean of corrupted sample $\tilde{\mathbf{x}}_i$ on dimension $d$ is always $x_{id}$ itself as the corrupting distributions in this paper are unbiased. This assumption is necessary as biases in the corrupting distribution may lead to undesired difficulty in selection of corrupting distribution parameter $\eta_d$. Table 1 gives an overview of some corrupting distributions.

However, the marginalized cross-view learning loss function in Eq.(5) has a trivial solution when $\mathbf{B}=0$ and $\mathbf{W}=0$, indicating that the current configuration is underconstrained. It is necessary to incorporate an additional regularization on $\mathbf{B}$ to guide a reasonable solution.

## Marginalized Denoising Autoencoder Regularization

In this section, we introduce a marginalized denoising autoencoder regularization for estimating $\mathbf{B}$. Our intention is to enrich the incomplete user tags. In particular, the basic building block of our regularization framework is a one layer linear de-noising autoencoder. From a given set of input tag set, we sample infinite tag vectors with replacement. We corrupt these inputs by random tag removal: each tag is set to 0 with probability $p$. Then we reconstruct the original tag set from the corrupted versions with a mapping $\mathbf{B}$. If this corruption mechanism matches the true corruption mechanism, then we can re-apply denoising transformation $\mathbf{B}$ to $\mathbf{y}$ so that it is likely that a complete tag set can be recovered.

**Corruption** Our training of $\mathbf{B}$ is based on one crucial insight: if a tag already exists in some input tag vector $\mathbf{y}$, $\mathbf{B}$ should be able to predict it from the remaining labels in $\mathbf{y}$. We therefore create a corrupted tag set by removing each tag in $\mathbf{y}$ with probability $p\geq 0$. In particular, for each user tag vector $\mathbf{y}$ and dimension $t$, $p(\tilde{y}_t=0)=p$ and $p(\tilde{y}_t=y_t)=1-p$.

**Reconstruction** A mapping $\mathbf{B}$ is then learned to reconstruct the original tag vector $\mathbf{y}$ from the corrupted version $\tilde{\mathbf{y}}$ by minimizing the squared reconstruction error,

$$\mathbf{B}^*=\arg\min_{\mathbf{B}}\frac{1}{n}\sum_{i=1}^{n}||\mathbf{y}_i-\mathbf{B}\tilde{\mathbf{y}}_i||^2 \tag{6}$$

Here, $\mathbf{B}$ can be considered as a regression matrix that predicts the presence of tags given the existing tags in $\tilde{\mathbf{y}}$. Further, to reduce variance in $\mathbf{B}$, we take repeated samples of

$\tilde{\mathbf{y}}$. In the limit (with infinitely corrupted versions of $\mathbf{y}$), the expected loss function under the corrupting distribution can be expressed as

$$\begin{aligned}\mathcal{R}(\mathbf{B}) =& \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[||\mathbf{y}_i - \mathbf{B}\tilde{\mathbf{y}}_i||^2\right]_{p(\tilde{\mathbf{y}}_i|\mathbf{y}_i)} \\ =& \frac{1}{n}\text{trace}\left(\mathbf{B}\mathbf{Q}_y\mathbf{B}^\top - 2\mathbf{P}_{yy}\mathbf{B}^\top + \mathbf{Y}\mathbf{Y}^\top\right) \end{aligned} \quad (7)$$

where $\mathbf{P}_{yy} = \sum_{i=1}^{n}\mathbf{y}_i\mathbb{E}\left[\tilde{\mathbf{y}}_i\right]^\top$, $\mathbf{Q}_y = \sum_{i=1}^{n}\mathbb{E}[\tilde{\mathbf{y}}_i]\mathbb{E}\left[\tilde{\mathbf{y}}_i\right]^\top + \mathbf{V}\left[\tilde{\mathbf{y}}_i\right]$, and

$$[\mathbf{Q}_\mathbf{y}]_{\alpha,\beta} = \begin{cases}\mathbf{S}_{\alpha\beta}\mathbf{q}_\alpha\mathbf{q}_\beta & \text{if } \alpha \neq \beta \\ \mathbf{S}_{\alpha\beta}\mathbf{q}_\alpha & \text{if } \alpha = \beta\end{cases} \quad (8)$$

$$[\mathbf{P}_{yy}]_{\alpha\beta} = \mathbf{S}_{\alpha\beta}\mathbf{q}_\beta \quad (9)$$

where $\mathbf{q}_\alpha = \mathbf{q}_\beta = 1 - p$ and $\mathbf{S} = \mathbf{Y}\mathbf{Y}^\top$ is the covariance matrix of the uncorrupted tag set.

## Regularized Marginalized Cross-View Learning

In this section we combine the marginalized corrupted features framework in Eq.(4) with the marginalized denoising autoencoder regularization in Eq.(7) to solve the problem of learning with marginalized corrupted features and labels together. The joint loss function can be written as follows:

$$\begin{aligned}\mathcal{J}\left(\mathbf{B},\mathbf{W};\mathbf{x},\mathbf{y}\right) =& \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[||\mathbf{B}\mathbf{y}_i - \mathbf{W}\tilde{\mathbf{x}}_i||^2\right]_{p(\tilde{\mathbf{x}}_i|\mathbf{x}_i)} \\ &+ \gamma\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[||\mathbf{y}_i - \mathbf{B}\tilde{\mathbf{y}}_i||^2\right]_{p(\tilde{\mathbf{y}}_i|\mathbf{y}_i)}\end{aligned} \quad (10)$$

*Regularized marginalized cross-view learning* is referred to minimizing the above objective function. The first term of the right-hand side in Eq.(10) is the *marginalized cross-view learning*, which treats sample features and tags as multi-view data and attempts to find optimal transformations that help obtain the best alignment between data from two different views. The second term is the *marginalized denoising autoencoder regularization*, which ensures that a reliable enrichment mapping $\mathbf{B}$ can be obtained.

**Optimization and Extensions** The loss in Eq.(10) can be efficiently optimized using coordinate descent. When $\mathbf{B}$ is fixed, the computation of regression matrix $\mathbf{W}$ reduces to the problem in Eq.(5) and can be solved in closed form:

$$\mathbf{W} = \mathbf{B}\mathbf{P}_{xy}\mathbf{Q}_x^{-1} \quad (11)$$

where $\mathbf{P}_{xy}$ and $\mathbf{Q}_x$ can be computed analytically following the definition and the statistical properties of the corrupting distributions in Table 1.

When $\mathbf{W}$ is fixed, the problem in Eq.(10) with respect to $\mathbf{B}$ can be reformulated as

$$\begin{aligned}\mathcal{R}\left(\mathbf{B}\right) =& \frac{1}{n}\text{trace}\left(\mathbf{B}\mathbf{Y}\mathbf{Y}^\top\mathbf{B}^\top - 2\mathbf{B}\mathbf{P}_{xy}\mathbf{W}^\top\right) \\ &+ \gamma\frac{1}{n}\text{trace}\left(\mathbf{B}\mathbf{Q}_y\mathbf{B}^\top - 2\mathbf{P}_{yy}\mathbf{B}^\top + \mathbf{Y}\mathbf{Y}^\top\right)\end{aligned} \quad (12)$$

Table 2: Statistics of the three data sets.

| Data set | Examples | Labels | Attributes |
|---|---|---|---|
| Bibtex | 7395 | 159 | 1836 |
| Bookmarks | 20000 | 208 | 2150 |
| Enron | 1702 | 53 | 1001 |

Similarly, when $\mathbf{W}$ is fixed, the solution of $\mathbf{B}$ in Eq.(12) can be solved as follows:

$$\mathbf{B} = \left(\gamma\mathbf{P}_{yy} + \mathbf{W}\mathbf{P}_{xy}^\top\right)\left(\gamma\mathbf{Q}_y + \mathbf{Y}\mathbf{Y}^\top\right)^{-1} \quad (13)$$

where $\mathbf{P}_{yy}$ and $\mathbf{Q}_y$ can be computed analytically following Eq.(9).

Importantly, the optimal mapping $\mathbf{B}$ can be derived under a closed form and computed efficiently. Such conclusion holds for most corrupting models with exponential-family distribution. The loss in Eq.(10) is jointly convex with respect to $\mathbf{B}$ and $\mathbf{W}$. Consequently, it is guaranteed that the coordinate descent converges to the global minimum.

Based on the above analysis, the algorithm of RMCV is outlined in Algorithm 1.

**Algorithm 1:** RMCV Algorithm

**Input :** Data sets $\mathbf{X}$, $\mathbf{Y}$.
**Output:** Estimated mappings $\mathbf{B}$ and $\mathbf{W}$.
**Test :** Given an image $\mathbf{x}$, $\mathbf{W}\mathbf{x}$ is used to score the dictionary of tags

Choose the corrupting distribution for $\mathbf{X}$; obtain $\mathbf{P}_{xy}$ and $\mathbf{Q}_x$ with Eq.(5);
Choose the dropout probability $p$; obtain $\mathbf{P}_{yy}$ and $\mathbf{Q}_y$ with Eq.(9);
**Repeat**
  Optimize $\mathbf{W}$ with Eq.(11);
  Optimize $\mathbf{B}$ with Eq.(13);
**until** *Convergence*;

**Stacked Tag Enrichment** A key component of the success of denoising autoencoder is the fact that they can be stacked to create a "deep architecture". With the marginalized denoising autoencoder regularization, our framework also has the same capability. We stack several layers by feeding the enriched tag representation of the $t^{th}$ layer as the input to the $(t+1)^{th}$ layer. In particular, with the enriched vector $\mathbf{B}\mathbf{y}_i$ as the tag representation for the $i$-th image, we optimize another layer of $\mathcal{J}\left(\mathbf{B}',\mathbf{W}';\mathbf{x},\mathbf{B}\mathbf{y}\right)$ to obtain new mappings $\mathbf{B}'$, $\mathbf{W}'$. To find the optimal number of the stacked layers, we perform model selection on a hold-out validation set, adding layers until the F1 score cannot be improved.

# Experiments

We evaluate RMCV on three standard multi-label text benchmark data sets. All data sets are obtained from http://mulan.sourceforge.net/datasets-mlc.html.

## Experimental Setup

We begin with a detailed descriptions of data sets, evaluation metrics, parameter setup, and baselines.

**Datasets** We have used three multi-label datasets, namely Bibtex, Bookmarks, and Enron for experimentation purpose. Their statistics is described in Table 2.

**Bibtex** & **Bookmarks** data sets are from Bibsonomy[1]. Bibsonomy is a social bookmarking and publication sharing system; it allows users to store and organize Bookmark (web pages) and Bibtex entries. Tagging is the main provided tool for content management in BibSonomy. Users can freely assign tags to Bookmarks or Bibtex items when they submit the items to the system. In particular, Bibtex data set contains meta data for the bibtex items such as the title of a paper and the authors. Bookmarks contain metadata for bookmark items such as the URL of the web page and a description of the web page.

**Enron** dataset contains email messages. It is from Enron corpus and is made public during the legal investigation concerning the Enron corporation. For this experiment, we use a subset of about 1700 labeled email messages.

**Evaluation Metric** Three metrics, precision, recall, and F1 score, are often used to measure the performance of a tagging algorithm. Here, we also use them as our evaluation metrics. First, all the text data are labeled with the five most relevant tags (i.e., tags with the highest prediction value). Second, precision (P) and recall (R) are computed for each tag. The reported measurements are the average across all the tags. Further, both factors are combined in F1 score ($F1 = 2\frac{P*R}{P+R}$), which is reported separately. In all the metrics a higher value indicates a better performance.

**Setup** We use cross-validation to estimate the performance of different methods. On the Bibtex and Enron data sets, we follow the experimental setup used in **Mulan**[2]. Since there is no fixed split in the Bookmarks data set in **Mulan**, we use a fixed training set of 80% of the data, and evaluate the performance of our predictions on the fixed test set of 20% of the data. We follow the setup of (Chen, Zheng, and Weinberger 2013) and weigh each example in a tf-idf-like fashion to give more weight on the losses from rare tags during training.

**Baselines** To demonstrate how RMCV improves the tagging performance in comparison with the state-of-the-art tagging methods, we compare it with the following representative tagging methods from the recent literature:

- LeastSquare (Bishop 2006).
- FastTag, a model which uses marginalized denosing autoencoder regularization (Chen, Zheng, and Weinberger 2013).
- Marginalized corrupted feature with blankout corruption (MCF (Blankout) in short) (van der Maaten et al. 2013).
- Marginalized corrupted feature with Poisson corruption (MCF (Poisson) in short) (van der Maaten et al. 2013).

In addition, we also study various different configurations of the proposed algorithm:

- RMCV (Blankout): using the blankout distribution to corrupt features.
- RMCV (Poisson): using the Poisson distribution to corrupt features.

In particular, when we adopt the Gaussian distribution as the corrupting distribution, RMCV reduces to FastTag.

## Experimental Results

In Table 3, we summarize the precision, recall, and F1 score of the Bibtex, Bookmarks, and Enron data sets, for LeastSquare, MCF (Blankout), MCF (Poisson), FastTag, RMCV (Poisson), and RMCV (Blankout), respectively. On the task of multi-label text tagging, compared with RMCV, LeastSquare uses the limited training example set as the training set and mistakenly takes the incomplete training tag set as the complete training tag set. Although MCF exploits the marginalized corrupted features, it ignores the influence of the incomplete training tag set. FastTag considers the training set as incomplete tagged dataset, but it cannot take advantage of the marginalized corrupted features to generalize well to the test data. Consequently, from Table 3, we see that RMCV performs better than leastSquare, MCF, and FastTag on the task of multi-label text tagging as the F1 scores achieved by RMCV are much higher than those achieved by the competing models in most cases. In particular, RMCV improves over the competing models for both blankout corruption and Poisson corruption in most cases. The best performance tends to be achieved by RMCV with blankout corruption with high corruption levels, i.e., when $q$ is at about $0.8$. In addition, part of the recall scores achieved by RMCV are lower than those achieved by the competing models. For the decreased recall values, we suppose this may be caused by the label completion procedure, which may increase the false negative rate while significantly decreasing the false positive rate.

The experiments also reveal a number of interesting observations:

- Both RMCV and MCF models obtain a better performance than the LeastSquare. This shows the importance of considering learning with marginalized corrupted features.
- Both RMCV and FastTag models outperform the LeastSquare. This indicates that learning with marginalized corrupted labels helps improve the generalization performance on the problem of multi-label text tagging.
- RMCV performs better than MCF and FastTag. This shows that combining the marginalized corrupted features and marginalized corrupted labels simultaneously may lead to a more robust method for multi-label learning.

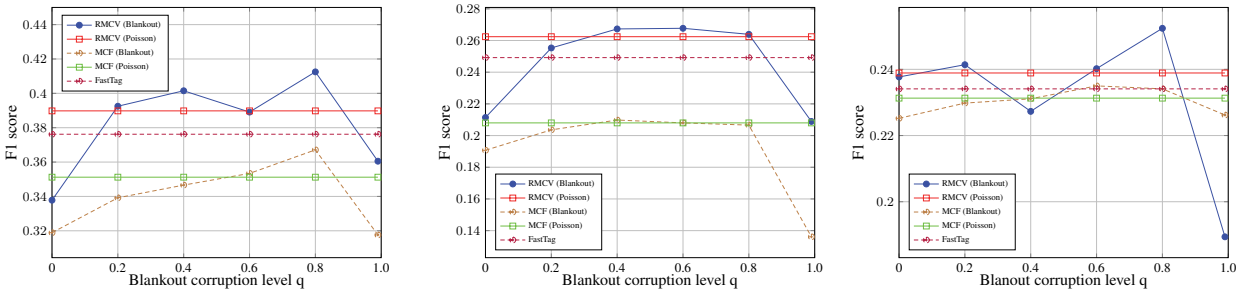
Figure 1(a), Figure 1(b), and Figure 1(c) show the test F1 scores of FastTag, MCF (Blankout), MCF (Poisson), RMCV (Blankout), and RMCV (Poisson) as a function of the blankout corruption level $q$ on Bibtex, Bookmarks, and Enron data sets, respectively. Herein, corruption level $q = 0$ corresponds to a Gaussian corruption on features with non-bias mean value and zero variance. The results show: 1) that RMCV improves over standard predictors for both blankout

[1] http://www.bibsonomy.org

[2] http://mulan.sourceforge.net/datasets-mlc.html

Table 3: Comparison of RMCV and the competing models in terms of precision, recall, and F1 score on the three data sets.

| | Bibtex | | | Bookmarks | | | Enron | | |
|---|---|---|---|---|---|---|---|---|---|
| Methods | precision | recall | F1 | precision | recall | F1 | precision | recall | F1 |
| LeastSquares | 0.2499 | 0.4975 | 0.3326 | 0.1439 | 0.2932 | 0.1931 | 0.2144 | 0.2396 | 0.2263 |
| MCF (Blankout) | 0.2896 | 0.5013 | 0.3671 | 0.1576 | **0.3136** | 0.2098 | 0.2084 | 0.2565 | 0.2301 |
| MCF (Poisson) | 0.2641 | **0.5240** | 0.3512 | 0.1572 | 0.3071 | 0.2080 | 0.2153 | 0.2500 | 0.2313 |
| FastTag | 0.3594 | 0.3948 | 0.3762 | 0.2439 | 0.2547 | 0.2492 | 0.2193 | 0.2512 | 0.2342 |
| RMCV (Poisson) | 0.3691 | 0.4128 | 0.3898 | 0.2459 | 0.2813 | 0.2624 | 0.2217 | **0.2591** | 0.2389 |
| RMCV (Blankout) | **0.3986** | 0.4274 | **0.4125** | **0.2517** | 0.2895 | **0.2693** | **0.2583** | 0.2468 | **0.2524** |




(a) Bibtex. (b) Bookmarks. (c) Enron.

Figure 1: The F1 score for the testing set as a function of blankout corruption level $q$ for FastTag, MCF (Poisson), MCF (Blankout), RMCV (Poisson), and RMCV (Blankout), respectively.

corruption and Poisson corruption on almost all the cases; 2) that RMCV with Poisson corruption leads to significant performance improvements over standard methods while introducing no additional hyperparameters.

From Figure 1(a), Figure 1(b), and Figure 1(c), we observe that the F1 scores of the testing set for RMCV and MCF models both increase when the corruption level increases at the start, which shows that it is helpful to use the marginalized corrupted features to improve the tagging performance. It is also noted that after a certain point when the the corruption level continues to increase, the performance may substantially drop. This is due to the *over corruption* issue, which loses much useful feature information.

Figure 2 demonstrates the comparison between RMCV and the competing models at different levels of tag sparsity of the training set. We gradually stack the training data into a larger tag sets, starting by giving each document only one tag (down sampled from the full tag set if more tags are available), then up to two tags, and so on. As we see from Figure 2, for training set with the maximum number of tags $n \in [1, 3]$, RMCV (Blankout) outperforms MCF (Blankout) with about 2% gain and FastTag with about 3% gain. With the maximum number of tags increases, RMCV (Blankout) outperforms the competing models with significant margins. This is due to the fact that the more numbers of tags make the marginalized denoising autoencoder regularization more effective and the learned tag enrichment mapping $\mathbf{B}$ more accurate. Although FastTag performs worse than MCF models when the maximum number of tags is small, its performance improves fast with the increase of the maximum number of tags and outperforms MCF models when the maximum

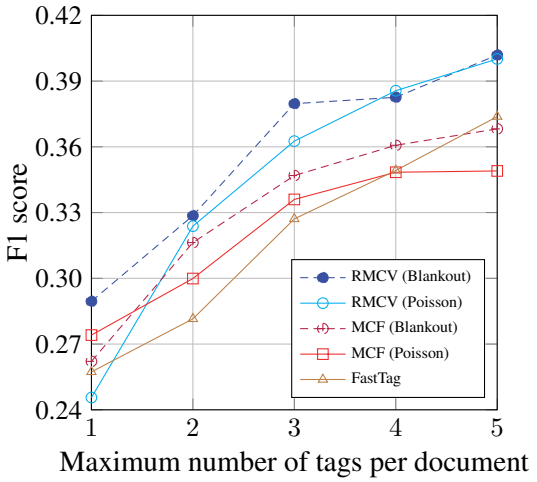



Figure 2: Performance in terms of F1 score as a function of the maximum number of tags provided for each training document on the Bibtex data set.

number of tags is larger. This also shows the importance of learning with marginalized corrupted labels.

## Conclusion

Tagging has become an important means responsible for many real-world applications noticeably including web applications. A robust tagging method must have the capability to meet the two challenging requirements: limited train-

ing samples and incomplete training labels. We have studied this challenging problem by learning with marginalized corrupted features and labels together and propose a discriminative model, called RMCV, that is trained with artificial feature noise and label noise through a regularized marginalized cross-view learning. While RMCV is a general method for learning tagging, in the evaluations we focus on the specific application of multi-label text tagging. Extensive evaluations on three benchmark data sets demonstrate that RMCV outstands with a superior performance in comparison with the state-of-the-art literature.

## Acknowledgments

This work was supported by NSF China (No. 61572111, 61433014, 61440036), a 973 project of china (No.2014CB340401), a 985 Project of UESTC (No.A1098531023601041) and Basic Research Projects of China Central Universities ( No. ZYGX2014J058, A03012023601042). Z. Zhang was also supported in part by the National Basic Research Program of China (2012CB316400).

## References

Biggio, B.; Nelson, B.; and Laskov, P. 2011. Support vector machines under adversarial label noise. In *ACML*, 97–112.

Bishop, C. M. 1994. Training with noise is equivalent to tikhonov regularization. *Neural Computation* 7:108–116.

Bishop, C. M. 2006. *Pattern recognition and machine learning*. Springer.

Burges, C. J. C., and Schölkopf, B. 1996. Improving the accuracy and speed of support vector machines. In *NIPS*, 375–381.

Chechik, G.; Heitz, G.; Elidan, G.; Abbeel, P.; and Koller, D. 2008. Max-margin classification of data with absent features. *Journal of Machine Learning Research* 9:1–21.

Chen, M.; Xu, Z. E.; Weinberger, K. Q.; and Sha, F. 2012. Marginalized denoising autoencoders for domain adaptation. In *ICML*.

Chen, M.; Zheng, A. X.; and Weinberger, K. Q. 2013. Fast image tagging. In *ICML*, 1274–1282.

Duda, R. O.; Hart, P. E.; and Stork, D. G. 2001. *Pattern Classification (2nd Ed)*. Wiley.

Elisseeff, A., and Weston, J. 2001. A kernel method for multi-labelled classification. In *NIPS*, 681–687.

Globerson, A., and Roweis, S. T. 2006. Nightmare at test time: robust learning by feature deletion. In *ICML*, 353–360.

Hinton, G. E.; Srivastava, N.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *CoRR* abs/1207.0580.

Hsu, D.; Kakade, S.; Langford, J.; and Zhang, T. 2009. Multi-label prediction via compressed sensing. In *AAAI*, 772–780.

Lin, C. F., and de Wang, S. 2004. Training algorithms for fuzzy support vector machines with noisy data. *Pattern Recognition Letters* 25:1647–1656.

Liu, W., and Tsang, I. W. 2015. Large margin metric learning for multi-label prediction. In *AAAI*, 2800–2806.

Liu, Y.; Jin, R.; and Yang, L. 2006. Semi-supervised multi-label learning by constrained non-negative matrix factorization. In *AAAI*, 421–426.

Sietsma, J., and Dow, R. J. F. 1991. Creating artificial neural networks that generalize. *Neural Network.* 4:67–79.

Srivastava, N.; Hinton, G. E.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15(1):1929–1958.

Teo, C. H.; Globerson, A.; Roweis, S. T.; and Smola, A. J. 2007. Convex learning with invariances. In *NIPS*, 1489–1496.

van der Maaten, L.; Chen, M.; Tyree, S.; and Weinberger, K. Q. 2013. Learning with marginalized corrupted features. In *ICML*, 410–418.

Van Hulse, J., and Khoshgoftaar, T. M. 2006. Class noise detection using frequent itemsets. *Intelligent Data Analysis* 10:487–507.

Vincent, P.; Larochelle, H.; Bengio, Y.; and Manzagol, P.-A. 2008. Extracting and composing robust features with denoising autoencoders. In *ICML*, 1096–1103.

Wager, S.; Wang, S. I.; and Liang, P. 2013. Dropout training as adaptive regularization. In *NIPS*, 351–359.

Wang, C.; Jing, F.; Zhang, L.; and Zhang, H.-J. 2007. Content-based image annotation refinement. In *CVPR*, 1–8.

Webb, A. R. 1994. Functional approximation by feed-forward networks: a least-squares approach to generalization. *IEEE Transactions on Neural Networks* 5(3):363–371.

Xu, H.; Caramanis, C.; and Mannor, S. 2009. Robustness and regularization of support vector machines. *Journal of Machine Learning Research* 10:1485–1510.

Yu, K.; Yu, S.; and Tresp, V. 2005. Multi-label informed latent semantic indexing. In *SIGIR*, 258–265.

Zhang, M., and Zhou, Z. 2007. ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition* 40(7):2038–2048.

Zhang, M., and Zhou, Z. 2014. A review on multi-label learning algorithms. *IEEE Trans. Knowl. Data Eng.* 26(8):1819–1837.

Zhu, X., and Wu, X. 2004. Class noise vs. attribute noise: A quantitative study. *Artif. Intell. Rev.* 22(3):177–210.