# Learning from Incomplete Image Tags

**Minmin Chen** **Kilian Q. Weinberger**
Washington University in St. Louis
Saint Louis, MO 63130
mchen, kilian@cse.wustl.edu

**Alice X. Zheng**
Microsoft Research
Redmond, WA 98052
alicez@microsoft.com

## Abstract

Obtaining high-quality training labels for learning can be an onerous task. In this paper, we look at the task of automatic image annotation, trained with only *partial* supervision. We propose MARCO, a novel algorithm that learns to predict the *complete* tag set of an image with the help of an auxiliary task that recovers the semantic relationship between tags. We formulate this as a convex programming problem and present an efficient optimization routine that iterates between two closed-form solution steps. We demonstrate on two real datasets that our approach out performs competitors, especially with very sparsely labeled training images.

## 1 Introduction

As technology enables data collection at ever increasing scale and speed, data analysts are looking at machine learning as the tool of choice. However, successful applications of machine learning often rely on the existence of large quantities of labeled examples, which can be difficult to obtain. Acquiring good labels can be time consuming and require domain expertise that few possess. In recent years, the research community has delved into crowdsourcing as a potential platform for acquiring labels cheaply and at scale [9]. However, crowdsourced labels can be inconsistent, and obtaining high-quality labels can still be expensive, especially for large-scale datasets.

In this paper, we present a novel method for machine-aided labeling. Namely, we reduce a complex labeling task to a much simpler one through the use of learning algorithms. We take image annotation [1, 4, 5, 8, 10, 11, 12] as an example of such a complex labeling task. Given an image, the goal is to annotate it with the complete list of tags that describe all visual features present in the image. Note that it is easy to tag an image with a few of the most prominent visual features, but to obtain the complete list can be quite difficult. The ESP game [15] takes the novel approach of allowing free-form input from the user, which is quick to do, but incentivizes pairs of labelers to match their answers. This results in tag sets with high precision, but without guarantees for high recall; each image may be tagged with only a small set of tags that describe the most obvious visual features. To alleviate the need for completely labeling a large set of training images, several existing works resort to semi-supervised approaches to leverage unlabeled or weakly labeled data from the web [6, 13, 14]. In this work, we explore another aspect of partial supervision for image annotation.

We present Marginalized Co-regularization (MARCO), a novel algorithm for image annotation that uses a simple yet effective trick to cope with overly sparse supervision: It treats its training data (images with partial tags) as *unlabeled multi-view* data. It then follows the spirit of co-training [2] and learns *two* classifiers to predict tag annotations: the first uses as input the image features and learns the mapping we hope to learn; the second uses the existing tags and is a purely auxiliary learning task. We propose a *joint* convex loss function that combines both classifiers via co-regularization and forces them to agree with their annotations. Our loss function can be trained efficiently through alternating optimization with simple closed-form updates. We demonstrate on real world data sets that MARCO outperforms several competitive baselines and is particularly strong in scenarios with very limited supervision.

## 2 Method

Let $\mathcal{T} = \{\omega_1, \cdots, \omega_T\}$ denote the dictionary of annotation tags of size $T$. Suppose we are given a *partial* training data $D = \{(\mathbf{x}_1, \mathbf{y}_1), \cdots, (\mathbf{x}_n, \mathbf{y}_n)\} \subset \mathcal{R}^d \times \{0,1\}^T$, where each $\mathbf{y}_i$ is a small partial subset of tags that are appropriate for image $\mathbf{x}_i$. Our goal is to learn a function $\mathbf{W} : \mathcal{R}^d \to \mathcal{T}$ which maps from an image $\mathbf{x}$ to its *complete* tag set. As we are only provided with an incomplete set of tags, we create an additional auxiliary problem and obtain two sub-tasks: 1. train an image classifier $\mathbf{x}_i \to \mathbf{W}\mathbf{x}_i$, which predicts the complete tag set from image features. 2. train a mapping $\mathbf{y}_i \to \mathbf{B}\mathbf{y}_i$ to *enrich* the existing overly sparse tag vector $\mathbf{y}_i$ by estimating which tags are likely to co-occur with those already in $\mathbf{y}_i$. As we lack the exact labels to perform supervised learning, we instead train both classifiers simultaneously and force them to agree, minimizing

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

Clearly, the loss in Eq. (1) has a trivial solution at $\mathbf{B} = \mathbf{0} = \mathbf{W}$. Hence, to avoid uninteresting solutions, we add a carefully chosen regularization term to guide one of the two mappings ($\mathbf{B}$) into a useful direction.

**Marginalized blank-out regularization.** To avoid trivial solutions, we add a regularizer that ensures that the mapping $\mathbf{B}$ performs well at predicting likely tags (inspired by marginalized Stacked Denoising Autoencoders (mSDA) [3]). The intention behind $\mathbf{B} : \{0,1\}^T \to \mathcal{R}^T$ is to enrich the incomplete user tags by turning on relevant keywords, which should have been tagged but were ignored by the human annotators. One way to learn this mapping is to artificially create a supervised dataset from the incomplete user tags. To this end, we randomly remove tags from the existing set and train $\mathbf{B}$ to predict which tags were removed. More formally, for each $\mathbf{y}$, a corrupted version $\tilde{\mathbf{y}}$ is created by randomly removing (i.e., setting to zero) each entry in $\mathbf{y}$ with some probability $p \geq 0$, *i.e.*, for each user tag vector $\mathbf{y}$ and dimensions $t$, $p(\tilde{y}_t = 0) = p$ and $p(\tilde{y}_t = y_t) = 1 - p$. The mapping $\mathbf{B}$ is then trained to reconstruct the original tag vector from its corrupted version,

$$\mathbf{B} = \operatorname*{argmin}_{\mathbf{B}} \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{y}_i - \mathbf{B}\tilde{\mathbf{y}}_i\|^2.$$

Here, each row of $\mathbf{B}$ is an ordinary least squares regressor that predicts the presence of a tag given all existing tags in $\tilde{\mathbf{y}}$. To reduce variance in $\mathbf{B}$, we take repeated samples of $\tilde{\mathbf{y}}$. In the limit (with infinitely many corrupted versions of $\mathbf{y}$), the expected reconstruction error can be expressed as

$$r(\mathbf{B}) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[\|\mathbf{y}_i - \mathbf{B}\tilde{\mathbf{y}}_i\|^2\right]_{p(\tilde{\mathbf{y}}_i)}. \tag{2}$$

**Joint loss function.** The joint loss combines the square loss in Eq. (1) with the mSDA regularization term $r(\mathbf{B})$ in Eq. (2) and the standard $l_2$ regularizer for $\mathbf{W}$,

$$\ell(\mathbf{B}, \mathbf{W}; \mathbf{x}, \mathbf{y}) = \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{B}\mathbf{y}_i - \mathbf{W}\mathbf{x}_i\|^2 + \frac{\gamma}{n}\sum_{i=1}^{n} \mathbb{E}\left[\|\mathbf{y}_i - \mathbf{B}\tilde{\mathbf{y}}_i\|^2\right]_{p(\tilde{\mathbf{y}}_i)} + \lambda\|\mathbf{W}\|_2^2. \tag{3}$$

The first term enforces that the tags enriched through co-occurrence with existing labels agree with the tags predicted by the content of the image. The second term ensures that the enrichment mapping $\mathbf{B}$ reliably predicts tags if they were to be removed from the training label set. Finally, the last term reduces the complexity of $\mathbf{W}$ and avoids overfitting.

**Optimization.** The loss in Eq. (3) can be efficiently optimized using coordinate descent. When $\mathbf{B}$ is fixed, the mapping $\mathbf{W}$ reduces to standard ridge-regression and can be solved for in closed form. Similarly, when $\mathbf{W}$ is fixed, the solution to Eq. (3) can be expressed as the well-known closed-form solution for ordinary least squares [3]:

$$\mathbf{B} = \left(\gamma\sum_{i=1}^{n} \mathbf{y}_i \mathbb{E}\left[\tilde{\mathbf{y}}_i\right]_{p(\tilde{\mathbf{y}}_i)}^{\top} + \sum_{i=1}^{n} \mathbf{W}\mathbf{x}_i\mathbf{y}_i^{\top}\right)\left(\gamma\sum_{i=1}^{n}\left(\mathbb{E}\left[\tilde{\mathbf{y}}_i\right]_{p(\tilde{\mathbf{y}}_i)} \mathbb{E}\left[\tilde{\mathbf{y}}_i\right]_{p(\tilde{\mathbf{y}}_i)}^{\top} + \mathbb{V}\left[\tilde{\mathbf{y}}_i\right]_{p(\tilde{\mathbf{y}}_i)}\right) + \sum_{i=1}^{n} \mathbf{y}_i\mathbf{y}_i^{\top}\right)^{-1}$$

where $\mathbb{E}[\tilde{\mathbf{y}}]_{p(\tilde{\mathbf{y}})}$ and $\mathbb{V}[\tilde{\mathbf{y}}]_{p(\tilde{\mathbf{y}})}$ denote the expected value and variance of the corruptions under the noise model. For the uniform "blank-out" noise introduced above, we have $\mathbb{E}[\tilde{\mathbf{y}}]_{p(\tilde{\mathbf{y}})} = (1-p)\mathbf{y}$, and $\mathbb{V}[\tilde{\mathbf{y}}]_{p(\tilde{\mathbf{y}})} = p(1-p)\text{diag}(\mathbf{y}^2)$. In other words, we can derive the optimal mapping $\mathbf{B}$ under closed form without explicitly creating any corruptions. The loss is jointly convex with respect to $\mathbf{B}$ and $\mathbf{W}$ and consequently coordinate descent converges to the global minimum.

**Bootstrapping to recover long-range tag associations.** The regularizer in Eq. (2) ensures that the mapping $\mathbf{B}$ can recover tags that are removed from the partial tag set. One way to interpret a partial tag set is to imagine that someone "removed" tags from the complete tag set. It is therefore intuitive to consider applying the learned mapping $\mathbf{B}$ on the labels $\mathbf{y}_1, \ldots, \mathbf{y}_n$ to obtain some "enriched" labels $\mathbf{y}'_1, \ldots, \mathbf{y}'_n$ and use these to re-optimize the joint loss in Eq. (3).[1] This bootstrapping can be applied arbitrarily often. In other words, we can stack up multiple layers of $(\mathbf{W}, \mathbf{B})$, using the enriched tags of the previous layer as input to the next layer. This has the advantage of recovering the semantic relationship between tags that do not directly co-occur with one another in the training set. At test time, given an image $\mathbf{x}$, the final mapping $\mathbf{W}^*$ is used to score the dictionary of tags. We refer to our algorithm as Marginalized Co-regularization (MARCO).

## 3 Experimental Results

We evaluate MARCO on two image annotation benchmark datasets.

*ESP game.*[2]. The dataset consists of 20,770 images of a wide variety, such as logos, drawings, and personal photos, collected in the ESP collaborative image labeling task [15]. Overall, the images are annotated with 268 tags. Each image is associated with a maximum of 15 and on average 4.6 tags. In our experiment, we use all the images with at least 7 tags (presumedly the set of images with relatively more complete list of tags) as the test set, and the remaining (16,748) images as the training set. Among the testing images, 1,000 of them are reserved for validation.

*IAPRTC-12.*[3]. The dataset consists of 19,627 images of sports, actions, people, animals, cities, landscapes and many other aspects of contemporary life [7]. Tags are extracted from the free-flowing text captions accompanying each image. Overall, 291 tags are used. On average, each image is annotated with 4.7 tags. Images with at least 8 tags are used for testing, while the remaining 15,276 images are used for training. Again, 1000 images out of the test set are reserved for validation.

For both datasets, we extract three types of visual descriptors, the global Gist features, local Sift and a robust Hue descriptors computed densely on a multi-scale grid. The three descriptors are normalized independently to have L1 unit length.

**Evaluation metric.** For each tag, we compute the 11-point interpolated average precision based on the output of each algorithm. The measurements are further averaged over all the tags to obtain the Mean interpolated Averaged Precision (MAP).

**Methods.** We compare against *leastSquare*, a ridge regression model, which uses the partial subset of tags $\mathbf{y}_1, \ldots, \mathbf{y}_n$ as labels to learn $\mathbf{W}$ in a supervised fashion. We also compare against the *Label Transfer* algorithm [12], which has been shown to outperform several existing multi-labeling methods despite its relative simplicity. To investigate the advantages of jointly learning the two classifiers $\mathbf{B}$ and $\mathbf{W}$ in MARCO, we also present the results of a version with separate optimization, in which the two sub-tasks are learned disjointly. It first learns the mapping $\mathbf{B}$ to enrich the tag set independently of the image features, minimizing Eq. (2); the enriched tags $\mathbf{B}\mathbf{y}_i$ are then used as labels for image classification in the second stage, where $\mathbf{W}$ is learned with ridge regression.

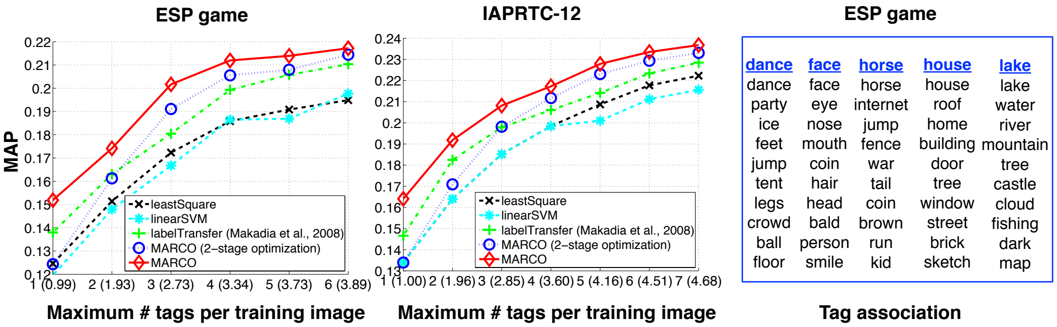
**Results.** Fig. 1 presents the test results on the ESP game (left) and IAPRTC-12 (middle). In order to compare the performance of each method at different levels of training set tag sparsity, we "stage" the training data into successively larger tag sets, starting by giving each image only one tag (subsampled if more tags are available), then two tags, and so on. Two observations can be made: 1. MARCO outperforms its competitors across all training settings, and 2. joint optimization of the two sub-tasks consistently outperforms separate optimization. By learning the two mappings simultaneously and forcing them to agree, MARCO offers a channel for these two mappings to "teach" each other with their own learned knowledge about the "ground truth" labels and to improve each other. Joint learning is particularly helpful when the training tag set is very sparse. In the extreme case

[1] The enriched tags $\mathbf{B}\mathbf{y}_i$ are real numbers. However, we do truncate $\mathbf{y}'_i$ to be within $[0,1]^T$.

[2] We use a subset out of the 60,000 images available at http://hunch.net/?p=23, which was also used in [8, 12]

[3] We used the same annotations as in [8, 12]

where each training image has only one tag, independent tag enrichment offers no gains because there are no observed tag correlations to build upon. MARCO, on the other hand, can be bootstrapped with the image features. Each column of Fig. 1 (right) shows five randomly chosen input tags (blue) and the top 10 tags (in decreasing order of their predicted value) that are reconstructed using the mapping $\mathbf{B}$. The table indicates a clear tendency that tags are associated with semantically similar tags.



| dance | face | horse | house | lake |
|---|---|---|---|---|
| dance | face | horse | house | lake |
| party | eye | internet | roof | water |
| ice | nose | jump | home | river |
| feet | mouth | fence | building | mountain |
| jump | coin | war | door | tree |
| tent | hair | tail | tree | castle |
| legs | head | coin | window | cloud |
| crowd | bald | brown | street | fishing |
| ball | person | run | brick | dark |
| floor | smile | kid | sketch | map |

Figure 1: Mean interpolated Averaged Precision (MAP) as a function of the maximum number of tags provided for each training image on the ESP game (left) and IAPRTC-12 (middle) datasets (The average number of tags per image is denoted in parenthesis.); (right) Tag reconstruction on the ESP game dataset.

**Effect of hyperparameter tuning.** We also examine the effect of the hyper-parameter $\gamma$, the number of bootstrap iterations. In Eq. (3), $\gamma$ prescribes the importance of the tag enrichment. As shown in Fig. 2 (left), the optimal value for $\gamma$ (tuned on the validation set) increases as more tags are provided. In other words, it is advantageous to rely more on the tag enrichment component as more labels become available (and the tag predictions become more accurate), whereas if labels are sparse it is better to rely more on the image feature mapping. Fig. 2 (middle) shows the performance of MARCO as we carry out multiple bootstrapping iterations. In general, performance improves as we bootstrap—although the effect saturates after about four iterations.

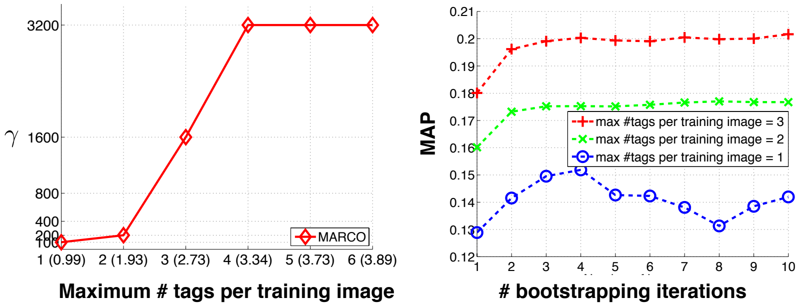


Figure 2: (left) optimal regularization factor $\gamma$ as a function of the maximum number of tags provided for each training image; (right) MAP as a function of bootstrapping iterations on the ESP game dataset. The three curves show different levels of tag sparsity.

**Computational time.** All experiments were conducted on a desktop with dual 6-core Intel i7 cpus with 2.66Ghz. For a total of around 16,748 training ($d = 3,812$) and 268 tags, MARCO took on average 4.89 seconds to train one bootstrap iteration. The optimal number of bootstrapping iterations (tuned on the validation set) ranges from 3 to 8 in different tag sparsity settings.

## 4 Conclusions

Our encouraging results suggest that there is merit in re-casting supervised classification problems with partial labels as unlabeled multi-view data. Our approach to jointly learn classifiers on each view with forced agreement works well in practice and gives rise to an algorithm, MARCO, which is based on a convex joint optimization problem that can be solved efficiently with coordinate descent. We hope our research will lead to interesting future work in many related learning scenarios.

## References

[1] D.M. Blei and M.I. Jordan. Modeling annotated data. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 127–134. ACM, 2003.

[2] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*. ACM, 1998.

[3] M. Chen, Z. Xu, K. Weinberger, and F. Sha. Marginalized denoising autoencoders for domain adaptation. In John Langford and Joelle Pineau, editors, *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, ICML '12, pages 767–774. ACM, New York, NY, USA, July 2012.

[4] R. Datta, D. Joshi, J. Li, and J.Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys (CSUR)*, 40(2):5, 2008.

[5] P. Duygulu, K. Barnard, J. De Freitas, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. *Computer VisionECCV 2002*, pages 349–354, 2006.

[6] R. Fergus, Y. Weiss, and A. Torralba. Semi-supervised learning in gigantic image collections. 2009.

[7] M. Grubinger, P. Clough, H. Müller, and T. Deselaers. The iapr tc-12 benchmark: A new evaluation resource for visual information systems. In *International Workshop OntoImage*, pages 13–23, 2006.

[8] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 309–316. Ieee, 2009.

[9] J. Howe. The rise of crowdsourcing. *Wired magazine*, 14(6):1–4, 2006.

[10] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 119–126. ACM, 2003.

[11] V. Lavrenko, R. Manmatha, and J. Jeon. A model for learning the semantics of pictures. In *NIPS*, 2003.

[12] A. Makadia, V. Pavlovic, and S. Kumar. A new baseline for image annotation. In *ECCV*, volume 8, pages 316–329, 2008.

[13] F. Schroff, A. Criminisi, and A. Zisserman. Harvesting image databases from the web. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[14] R. Socher and L. Fei-Fei. Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 966–973. IEEE, 2010.

[15] L. Von Ahn and L. Dabbish. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 319–326. ACM, 2004.