Fast Image Tagging

Minmin Chen MINMCHEN@AMAZON.COM
Amazon.com, Seattle, WA 98109

Alice Zheng ALICEZ@MICROSOFT.COM
Microsoft Research, Redmond, WA 98052

Kilian Q. Weinberger KILIAN@WUSTL.EDU
Washington University in St. Louis, St. Louis, MO 63130

Abstract

Automatic image annotation is a difficult and highly relevant machine learning task. Recent advances have significantly improved the state-of-the-art in retrieval accuracy with algorithms based on nearest neighbor classification in carefully learned metric spaces. But this comes at a price of increased computational complexity during training and *testing*. We propose FastTag, a novel algorithm that achieves comparable results with two simple linear mappings that are co-regularized in a joint convex loss function. The loss function can be efficiently optimized in closed form updates, which allows us to incorporate a large number of image descriptors cheaply. On several standard real-world benchmark data sets, we demonstrate that FastTag matches the current state-of-the-art in tagging quality, yet reduces the training and testing times by several orders of magnitude and has lower asymptotic complexity.

1. Introduction

Image tag annotations are an important component of searchable image databases such as Flickr^{TM}, Picassa^{TM} or Facebook^{TM}. However, a large fraction (over 50% in Flickr) of images have no tags at all and are hence never retrieved for text queries. Automatic image annotation is an essential tool towards surfacing this "dark content". A working image annotation engine can suggest tags to users (Weinberger et al., 2008) and thus increase the number of tagged images, or generate relevant tags for image retrieval directly.

Proceedings of the 30th International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28.

Automatic image annotation is a difficult machine learning task. Different type of objects require different image descriptors, *e.g.* rainbows can be identified through color histograms (Hafner et al., 1995), whereas insects can be best identified through local image descriptors (Lowe, 1999). Similar objects can look very different across images and may only be partially visible, thus necessitating large training data sets. Training labels are typically obtained through crowdsourcing and are noisy and notoriously incomplete. The ESP game (Von Ahn & Dabbish, 2004) proposes a solution to improve label quality by incentivizing pairs of labelers to match their answers. This results in tag sets with high precision but with no guarantees for high recall: each image may be tagged with only a small subset of tags that describe the most obvious visual features.

Recently, Makadia et al. (2008); Guillaumin et al. (2009) proposed new algorithms for automatic image annotation based on nearest neighbor methods. Guillaumin et al. (2009) carefully learn embeddings into metric spaces that combine a diverse set of image descriptors and assign tag-specific weights to overcome label sparsity. The resulting algorithm significantly improves over the prior state-of-the-art in both precision and recall. Although these approaches yield impressive results, they are impractical for large image databases with $n \gg 0$ images. Their training procedures scale on the order of $O(n^2)$. Moreover, the task of tagging a *single* test image is $O(n)$, *linear* with the training set size.

In real world applications, the number of images can be very large. Millions of images are added every day (*e.g.* 300 million images are uploaded to Facebook per day, with a total of 100 billion images[1]), rendering these methods impractical even to index the daily uploads.

[1]CNET 08/2012, http://tinyurl.com/9jfs7ut

In this paper, we present a novel learning algorithm for image tag annotation that achieves comparable accuracy to Guillaumin et al. (2009), but can be trained in $O(n)$ time and applied during testing in *constant time* w.r.t. the training set size. Our proposed algorithm, *FastTag*, can naturally incorporate many image descriptors and address the difficulties of label sparsity with a novel approach. It interprets its training data (images with partial tags) as *unlabeled multi-view* data and learns *two* classifiers to predict tag annotations: one attempts to reconstruct the (unknown) complete tag set from the few tags available during training; the other learns a mapping from image features to this reconstructed tag set. We propose a *joint* convex loss function that combines both classifiers via co-regularization and coerces them into agreement. Our loss function can be trained efficiently through alternating optimization with simple closed-form updates. We demonstrate on real world data sets that FastTag matches the highly competitive state-of-the-art in terms of precision and recall, but is several orders of magnitude faster during training and almost instantaneous during testing.

2. Related Work

In this section, we review some of the popular methods for automatic image annotation. The first group of methods are based on parametric topic models. Monay & Gatica-Perez (2004) extend the probabilistic latent semantic analysis model, and Barnard et al. (2003) extend the latent dirichlet allocation model to multimodal data. Each annotated image is modeled as a mixture of topics over visual and text features. The mixture proportions are shared between feature modes, but the topic distributions are distinct. The second group of methods (Jeon et al., 2003; Lavrenko et al., 2003; Feng et al., 2004) models the *joint* distribution of the image features and the tags with mixture models. The third group of methods trains discriminative models, such as SVM (Cusano et al., 2003), ranking SVM (Grangier & Bengio, 2008) and boosting (Hertz et al., 2004), to predict tags from image features.

While these methods achieve promising annotation results, their complex training processes limit the number of descriptors that can be incorporated. Recently proposed models such as the Joint Equal Contribution model of (Makadia et al., 2008) and the TagProp model of (Guillaumin et al., 2009) rely on local nearest neighborhoods and work surprisingly well despite their simplicity. TagProp is the current state-of-the-art method for image annotation. Its success can be attributed to three elements: 1. it incorporates a large number of different visual descriptors; 2. it can be trained effectively on images with incomplete tag sets; 3. it treats rare tags special.

Although Tagprop achieves superior performance on several benchmark datasets, the $O(n^2)$ training and $O(n)$ test complexity hinder its applicability to large scale datasets (where n is the number of examples in the training set). In this work, we introduce a new model that incorporates these three elements for successful annotation much more cheaply.

Most existing models assume that a complete list of relevant tags for each image is available at training time. However, in practice, this is either impractical or impossible for a large training set. It is much easier to tag an image with a few of the most prominent visual features than to obtain the complete list from a tag dictionary. To alleviate the need for complete labeling, several existing approaches (Fergus et al., 2009; Schroff et al., 2007; Socher & Fei-Fei, 2010) resort to semi-supervised approaches to leverage unlabeled or weakly labeled data from the web. We adopt the same assumption of sparse training tags and incorporate partial supervision in our work.

3. Method

We assume, as it is the case in real world applications, that only a few relevant tags are provided for each image during training. Given the training images annotated with incomplete tags, our goal is to learn a model that can infer the full list of tags from image features at test time. Our proposed algorithm is fast in training and almost instant prediction during testing (only a linear transformation is required). Thus we refer to our algorithm as *FastTag*.

Notation. Let $\mathcal{T} = \{\omega_1, \cdots, \omega_T\}$ denote the dictionary of T possible annotation tags. Let the training data be denoted by $D = \{(\mathbf{x}_1, \mathbf{y}_1), \cdots, (\mathbf{x}_n, \mathbf{y}_n)\} \subset \mathcal{R}^d \times \{0,1\}^T$, where each vector $\mathbf{x}_i \in \mathcal{R}^d$ represents the features extracted from the i-th image (for details see section 3.3 and section 4) and each $\mathbf{y}_i$ is a small *partial* subset of tags that are appropriate for the i-th image. Our goal is to learn a linear function $\mathbf{W} : \mathcal{R}^d \to \mathcal{T}$, which maps a test image $\mathbf{x}_i$ to its *complete* tag set.

3.1. Duo Classifier Formulation

In this section we introduce a new model for automatic image annotation from incomplete user tags. It jointly learns two classifiers on two sources, *i.e.*, image and text, to agree upon the list of tags predicted for each image. It leads to an optimization problem which is

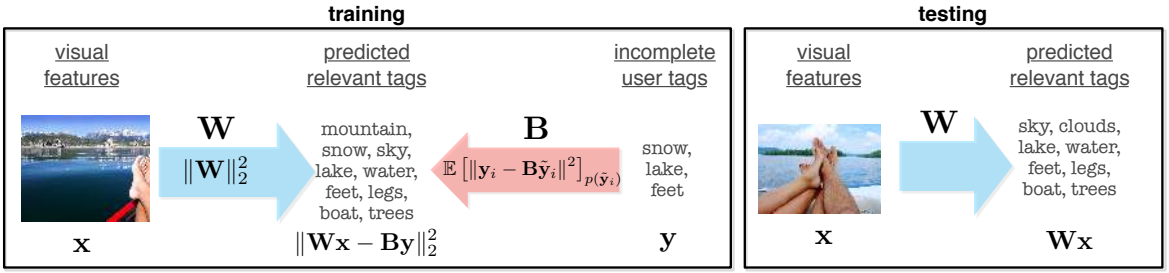

Figure 1. Schematic illustration of FastTag. During training two classifiers $\mathbf{B}$ and $\mathbf{W}$ are learned and co-regularized to predict similar results. At testing time, a simple linear mapping $\mathbf{x} \rightarrow \mathbf{W}\mathbf{x}$ predicts tags from image features.

jointly convex and has closed form solutions in each iteration of the optimization.

Co-regularized learning. As we are only provided with an incomplete set of tags, we create an additional auxiliary problem and obtain two sub-tasks: 1) training an image classifier $\mathbf{x}_i \rightarrow \mathbf{W}\mathbf{x}_i$ that predicts the complete tag set from image features, and 2) training a mapping $\mathbf{y}_i \rightarrow \mathbf{B}\mathbf{y}_i$ to *enrich* the existing sparse tag vector $\mathbf{y}_i$ by estimating which tags are likely to co-occur with those already in $\mathbf{y}_i$. We train both classifiers simultaneously and force their output to agree by minimizing

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

Here, $\mathbf{B}\mathbf{y}_i$ is the enriched tag set for the i-th training image, and each row of $\mathbf{W}$ contains the weights of a linear classifier that tries to predict the corresponding (enriched) tag based on image features.

The loss function as currently written has a trivial solution at $\mathbf{B}=\mathbf{0}=\mathbf{W}$, suggesting that the current formulation is underconstrained. We next describe additional regularizations on $\mathbf{B}$ that guides the solution toward something more useful.

Marginalized blank-out regularization. We take inspiration from the idea of marginalized stacked denoising autoencoders (Chen et al., 2012) and related works (**?**) in formulating the tag enrichment mapping $\mathbf{B}:\{0,1\}^T \rightarrow \mathcal{R}^T$. Our intention is to enrich the incomplete user tags by turning on relevant keywords that should have been tagged but were not. Imagine that the observed tags $\mathbf{y}$ are randomly sampled from the complete set of tags: it is a "corrupted" version of the original set. We leverage this insight and train the enrichment mapping $\mathbf{B}$ to reverse the corruption process. To this end, we construct a further corrupted version of the observed tags $\tilde{\mathbf{y}}$ and train $\mathbf{B}$ to reconstruct $\mathbf{y}$ from $\tilde{\mathbf{y}}$. If this secondary corruption mechanism matches the original corruption mechanism, then re-applying $\mathbf{B}$ to $\mathbf{y}$ would recover the likely original pristine tag set.

For simplicity, we use uniform corruption as the secondary corruption mechanism. In practice, human labelers may select tags with bias, not uniform probability. We can approximate the unknown corrupting distribution with piecewise uniform corruption in the learning step (see section 3.2). If prior knowledge on the original corruption mechanism is available, it can also easily be incorporated into our model.

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 $p \geq 0$ and therefore, for each user tag vector $\mathbf{y}$ and dimensions t, $p(\tilde{y}_t = 0) = p$ and $p(\tilde{y}_t = y_t) = 1 - p$. We train $\mathbf{B}$ to optimize

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

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 under the corrupting distribution can be expressed as

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

Let us denote as $\mathbf{Y} \equiv [\mathbf{y}_1, \cdots, \mathbf{y}_n]$ the matrix containing the partial labels for each image in each column. Define $\mathbf{P} \equiv \sum_{i=1}^{n}\mathbf{y}_i\mathbb{E}[\tilde{\mathbf{y}}_i]^\top$ and $\mathbf{Q} \equiv \sum_{i=1}^{n}\mathbb{E}[\tilde{\mathbf{y}}_i\tilde{\mathbf{y}}_i^\top]$, then we can rewrite the loss in (2) as

$$r(\mathbf{B}) = \frac{1}{n}\text{trace}(\mathbf{B}\mathbf{Q}\mathbf{B}^\top - 2\mathbf{P}\mathbf{B}^\top + \mathbf{Y}\mathbf{Y}^\top) \tag{3}$$

We use Eq. (3) to regularize $\mathbf{B}$. For the uniform "blank-out" noise introduced above, we have the expected value of the corruptions $\mathbb{E}[\tilde{\mathbf{y}}]_{p(\tilde{\mathbf{y}}|\mathbf{y})} = (1-p)\mathbf{y}$,

and the variance matrix $\mathbb{V}[\tilde{\mathbf{y}}]_{p(\tilde{\mathbf{y}}|\mathbf{y})} = p(1-p)\delta(\mathbf{y}\mathbf{y}^\top)$. Here $\delta(\cdot)$ stands for an operation that sets all the entries except the diagonal to zero (as we corrupt each tag independently, the variance matrix has non-zeros entries only on the diagonal). We can then compute the two matrices in Eq. (3) as

$$\begin{aligned} \mathbf{P} &= (1-p)\mathbf{Y}\mathbf{Y}^\top \\ \mathbf{Q} &= (1-p)^2\mathbf{Y}\mathbf{Y}^\top + p(1-p)\delta(\mathbf{Y}\mathbf{Y}^\top). \end{aligned} \tag{4}$$

Joint loss function. Combining the squared loss in Eq. (1) with the marginalized blank-out regularization term $r(\mathbf{B})$ in Eq. (3) and the standard ridge regression l_2 regularizer for $\mathbf{W}$, the joint loss function can be written as

$$\begin{aligned} \ell(\mathbf{B},\mathbf{W};\mathbf{x},\mathbf{y}) &= \underbrace{\frac{1}{n}\sum_{i=1}^{n}\|\mathbf{B}\mathbf{y}_i - \mathbf{W}\mathbf{x}_i\|^2 + \lambda\|\mathbf{W}\|_2^2}_{\textbf{Co-regularization}} \\ &+ \underbrace{\gamma r(\mathbf{B})}_{\textbf{Marginalized blank-out}}. \end{aligned} \tag{5}$$

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. A regularizer on $\mathbf{W}$ is included to reduce complexity and avoid overfitting. The last term ensures that the enrichment mapping $\mathbf{B}$ reliably predicts tags if they were to be removed from the training label set.

Test time. At test time, given an image $\mathbf{x}$, the final mapping $\mathbf{W}^*$ is used to score the dictionary of tags.

3.2. Optimization and Extensions

The loss in Eq. (5) can be efficiently optimized using block-coordinate descent. When $\mathbf{B}$ is fixed, the mapping $\mathbf{W}$ reduces to standard ridge-regression and can be solved for in closed form:

$$\mathbf{W} = \mathbf{B}\mathbf{Y}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T + n\lambda I)^{-1}, \tag{6}$$

where $\mathbf{X}$ and $\mathbf{Y}$ respectively contain the training image features and labels in columns.

Similarly, when $\mathbf{W}$ is fixed, the solution to Eq. (5) can be expressed as the well-known closed-form solution for ordinary least squares (Chen et al., 2012):

$$\mathbf{B} = \left(\gamma\mathbf{P} + \mathbf{W}\mathbf{X}\mathbf{Y}^\top\right)\left(\gamma\mathbf{Q} + \mathbf{Y}\mathbf{Y}^\top\right)^{-1}.$$

where $\mathbf{P}$ and $\mathbf{Q}$ can be computed analytically following eq. (4). In other words, we can derive the optimal mapping $\mathbf{B}$ under closed form without explicitly creating any corruptions. The conclusion holds for any corrupting models of which the expected value and variance can be computed analytically. The loss is jointly convex with respect to $\mathbf{B}$ and $\mathbf{W}$ and consequently coordinate descent converges to the global minimum. Fig. 1 contains a depiction of this algorithm.

Tag bootstrapping. The enrichment mapping $\mathbf{B}$ is trained to predict missing tags based on pairwise co-occurrence patterns. We would like to also reconstruct tags that do not co-occur together but tend to appear within similar contexts. As an example, the tag "pond" might rarely co-occur with "lake", as both describe similar things and annotators tend to use one or the other. However, it would be good to give the predictor $\mathbf{W}$ the flexibility to predict both from similar image features. We can achieve this via stacking: starting with the enriched vector $\mathbf{B}\mathbf{y}_i$ as the tag representation for the i-th image[2], we optimize another layer of $\ell(\mathbf{B}',\mathbf{W}';\mathbf{x},\mathbf{B}\mathbf{y})$ to obtain new mappings $\mathbf{B}',\mathbf{W}'$. We can have an arbitrary number of layers, each resulting in a new linear mapping $\mathbf{W}^t$ from image features to tags. To find the right trade-off between two much bootstrapping and too little, we perform model selection on a hold-out set, adding layers until it no longer improves the F1 measure.

Rare tags and Non-Uniform Corruption. Eq. (5) solves for the linear predictors $\mathbf{W}$ for all T tags simultaneously. This is computationally efficient in that it requires only one matrix inversion per iteration. However, it has the disadvantage that the prediction loss for each tag is weighed equally, which leads to the overall loss to be dominated by contributions from more frequent tags, sacrificing the prediction accuracy of rare tags. This is a known problem in tag prediction. Other approaches also find that dealing with rare tags is the key to improving tagging performance (Guillaumin et al., 2009). We introduce several re-optimization stages, where at each stage we solve a sub-problem of Eq. (5). That is, we identify a subset of tags with a recall below a certain threshold (in our experiments we set it to the average recall). We re-optimize (5) restricted to only the rows of $\mathbf{B}$ and $\mathbf{W}$ corresponding to such tags. We iterate until we no longer improve the F1 measure on a hold-out set.

This stage-wise re-optimization also allows us to approximate the unknown true corrupting distribution with piecewise estimates: each stage of re-optimization may set a different corruption probability p based on validation results on a hold-out set, keeping the corruption probability of remaining tags fixed at their previous values.

In addition, we weigh each example in a tf-idf-like fash-

[2]The enriched tags $\mathbf{B}\mathbf{y}_i$ are real numbers. When stacking, we truncate $\mathbf{B}\mathbf{y}_i$ to be within $[0,1]^T$.

ion so that losses from rare tags are given more weight during training. Specifically, each tag ω is assigned a cost $c_\omega = \frac{1}{n_\omega}$, where n_ω is the number of times tag ω appears in the training set. Thus, rarer tags are given a higher cost than the more frequent ones. We then assign each example a weight by simply summing over the costs of its active tags, so that examples with rarer tags contribute more to the loss in eq. (5). Let Λ denote an $n \times n$ diagonal matrix containing the weight for each training example, we can then solve for the optimal mapping as $\mathbf{W} = \mathbf{B}\mathbf{Y}\Lambda\mathbf{X}^T(\mathbf{X}\Lambda\mathbf{X}^T + n\lambda I)^{-1}$. The tag enrichment mapping $\mathbf{B}$ can be generalized to the weighted version in the same fashion.

3.3. Homogeneous feature mapping

Local kNN methods (Guillaumin et al., 2009; Makadia et al., 2008) enjoy the advantage of naturally identifying non-linear decision boundaries based on multiple feature spaces from different image features. In our work, we adopt linear image feature classifiers for their simplicity and speed, and instead incorporate non-linearity into the feature space as a preprocessing step. To this end, we adopt the homogeneous feature mapping method of Vedaldi and Zisserman (Vedaldi & Zisserman, 2012). For each visual descriptor $\mathbf{f}_m(\mathbf{x}) \in \mathcal{R}^{d_m}$ extracted from the input image, it uses an explicit feature mapping $\Psi_m : \mathcal{R}^{d_m} \rightarrow \mathcal{R}^{d_m(2r+1)}$ to project it to a slightly higher-dimensional feature space, in which the inner product approximates the kernel distance well. In other words, $\langle \Psi_m(\mathbf{f}_m(\mathbf{x})), \Psi_m(\mathbf{f}_m(\mathbf{x}'))\rangle \approx K_m(\mathbf{f}_m(\mathbf{x}), \mathbf{f}_m(\mathbf{x}'))$. For the family of additive kernels, such as the l_1-distance and χ^2-distance used in our experiments, the mapping $\Psi(\cdot)$ can be computed analytically and approximates the kernel well even with small r (in our experiment, we set $r = 1$). After projecting each visual descriptor independently, we further apply random projection (Vempala, 2005) to reduce the dimensionality[3].

4. Experimental Results

We evaluate FastTag[4] on three standard image annotation benchmark datasets. All data sets (with pre-extracted features) were obtained from `http://lear.inrialpes.fr/people/guillaumin/data.php`.

[3]The dimension k is roughly cross-validated using a least squares baseline.

[4]Our open source MATLAB^{TM} code is available for download at `http://www.cse.wustl.edu/~mchen/`.

4.1. Experimental Setup

We begin with a detailed description of the data sets, the visual descriptors and the evaluation metrics.

Corel5K. The dataset (Duygulu et al., 2006) contains 5,000 images collected from the larger Corel CD set. Each image is manually annotated with keywords from a dictionary of 260 distinct terms. On average, each image was annotated with 3.5 tags.

ESP game. The dataset consists of 20,770 imagesof a wide variety, such as logos, drawings, and personal photos, collected for the ESP collaborative image labeling task (Von Ahn & Dabbish, 2004). The images are annotated with a total of 268 tags. Each image is associated with a maximum of 15 and 4.6 tags on average.

IAPRTC-12.[5]. The dataset consists of 19,627 images of sports, actions, people, animals, cities, landscapes and many other aspects of contemporary life (Grubinger et al., 2006). Tags are extracted from the free-flowing text captions accompanying each image. Overall, 291 tags are used.

For all these datasets, we follow the training/test split used in previous work (Guillaumin et al., 2009; Makadia et al., 2008). Please refer to Guillaumin et al. (2009) for more detailed statistics on the datasets.

Feature extraction. We use the 15 different visual descriptors, extracted by Guillaumin et al. (2009) for each dataset. These include one Gist descriptor (Oliva & Torralba, 2001), six global color histograms, and eight local bag-of-visual-words features. As described in section 3.3, we adopt the explicit feature mapping of Vedaldi & Zisserman (2012) to obtain a non-linear feature transformation. Here we use the l_1 approximation (*i.e.* the Euclidean distance after the mapping approximates the l_1 distance) for the global color descriptors, and the approximated χ^2 distance for the local bag-of-visual-words features. Finally, we apply random projection after each feature mapping to reduce the dimensionality.

Evaluation metric. For full comparability, we adopt the same evaluation metrics as in Guillaumin et al. (2009). First, all image are annotated with the five most relevant tags (*i.e.* tags that have the highest prediction value). Second, precision (P) and recall (R) are computed for each tag. The reported measurements are averaged across all tags. For easier comparability, both factors are combined in the F1-score ($F1 = 2\frac{P*R}{P+R}$), which is reported separately. We also

[5]We used the same annotations as in (Guillaumin et al., 2009; Makadia et al., 2008)

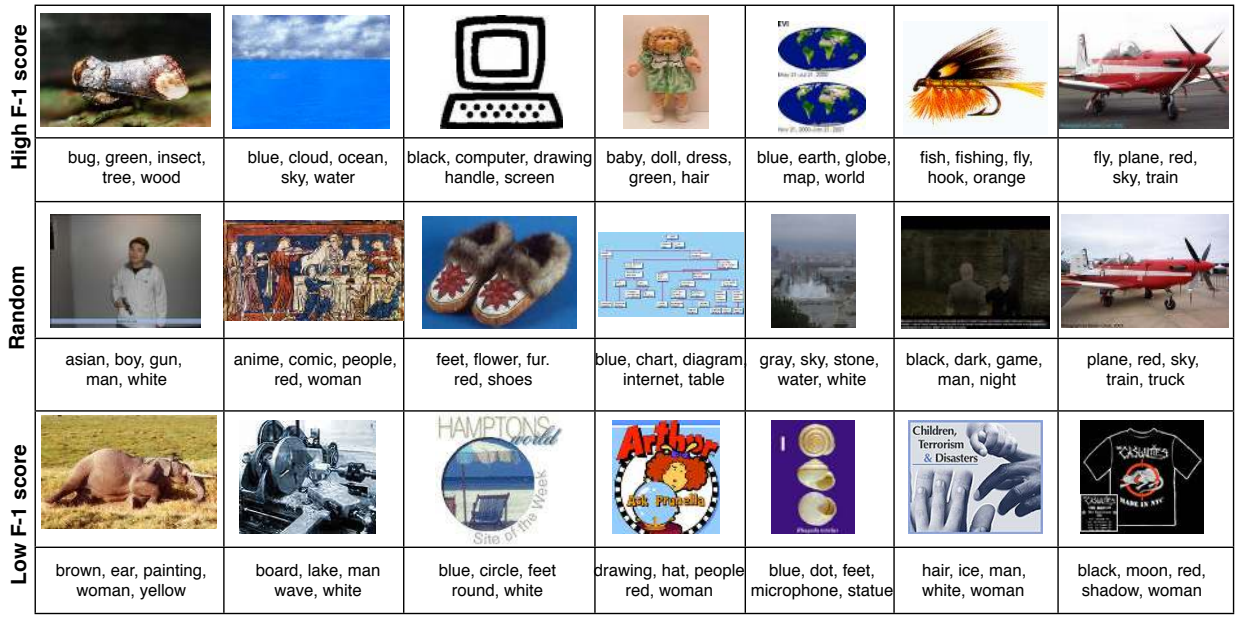

Figure 2. Predicted keywords using FastTag for sample images in the ESP game dataset (using all 268 keywords).

Table 1. Comparison of FastTag and TagProp in terms of P, R, F1 score and N+ on the Corel5K dataset. Previously reported results using other image annotation techniques are also included for reference.

Name	P	R	F1	N+
leastSquares	29	32	30	125
CRM (Lavrenko et al., 2003)	16	19	17	107
InfNet (Metzler & Manmatha, 2004)	17	24	20	112
NPDE (Yavlinsky et al., 2005)	18	21	19	114
SML (Carneiro et al., 2007)	23	29	26	137
MBRM (Feng et al., 2004)	24	25	24	122
TGLM (Liu et al., 2009)	25	29	27	131
JEC (Makadia et al., 2008)	27	32	29	139
TagProp (Guillaumin et al., 2009)	**33**	42	37	160
FastTag	32	**43**	37	**166**

report the number of keywords with non-zero recall value (N+). In all metrics a higher value indicates better performance.

Baselines. We compare against *leastSquares*, a ridge regression model which uses the partial subset of tags $\mathbf{y}_1, \ldots, \mathbf{y}_n$ as labels to learn $\mathbf{W}$, i.e., FastTag without tag enrichment. We also compare against the *TagProp* algorithm (Guillaumin et al., 2009), a local kNN method combining different distance metrics through metric learning. It is the current best performer on these benchmark sets. Most existing work do not provide publicly available implementations. As a result, we include their previously reported results for reference (Lavrenko et al., 2003; Metzler & Manmatha, 2004; Yavlinsky et al., 2005; Carneiro et al., 2007; Feng et al., 2004; Liu et al., 2009; Makadia et al., 2008) .

Table 2. Comparison of FastTag and TagProp in terms of P , R, F1 score and N+ on the Espgame and IAPRTC-12 datasets.

	ESP game				IAPR			
	P	R	F1	N+	P	R	F1	N+
leastSquares	35	19	25	215	40	19	26	198
MBRM	18	19	18	209	24	23	23	223
JEC	24	19	21	222	29	19	23	211
TagProp	39	**27**	**32**	238	45	**34**	**39**	**260**
FastTag	**46**	22	30	**247**	**47**	26	34	**280**

4.2. Comparison with related work

Table 1 shows a detailed comparison of FastTag to the leastSquares baseline and eight published results on the Corel5K dataset. We can make three observations: 1. The performance of FastTag aligns with that of TagProp (so far the best algorithm in terms of accuracy on this dataset), and significantly outperforms the other methods; 2. The leastSquares baseline, which corresponds to FastTag without the tag enricher, performs surprisingly well compared to existing approaches, which suggests the advantage of a simple model that can extend to a large number of visual descriptor, as opposed to a complex model that can afford fewer descriptors. One may instead more cheaply glean the benefits of a complex model via non-linear transformation of the features. 3. The duo classifier formulation of FastTag, which adds the tag enricher, alleviates the intrinsic label sparsity problem of image annotation. It leads to a 10% improvement on precision, 28% on recall, and an overall 20% improvement on F1 score over the leastSquares baseline. We also

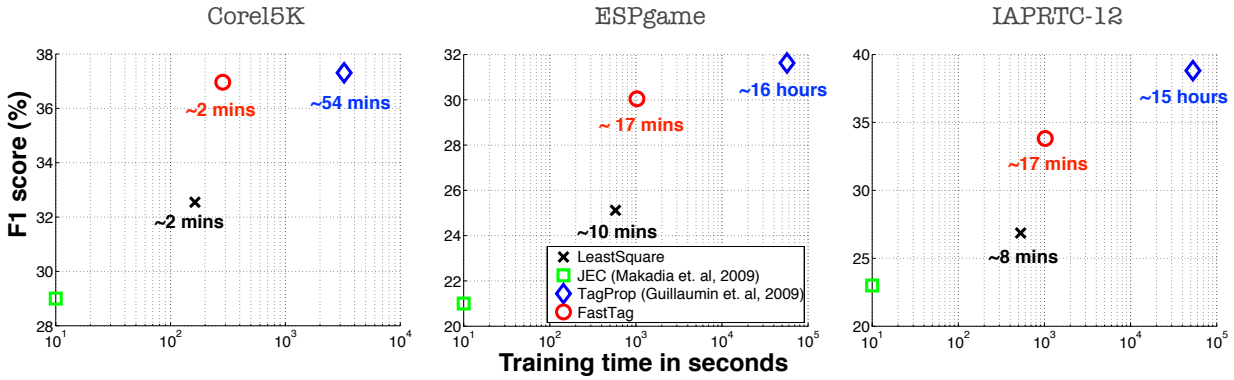

Figure 3. F1 score and training times on the three benchmark datasets. The graphs compare the results of FastTag with the leastSquares baseline and the TagProp algorithm.

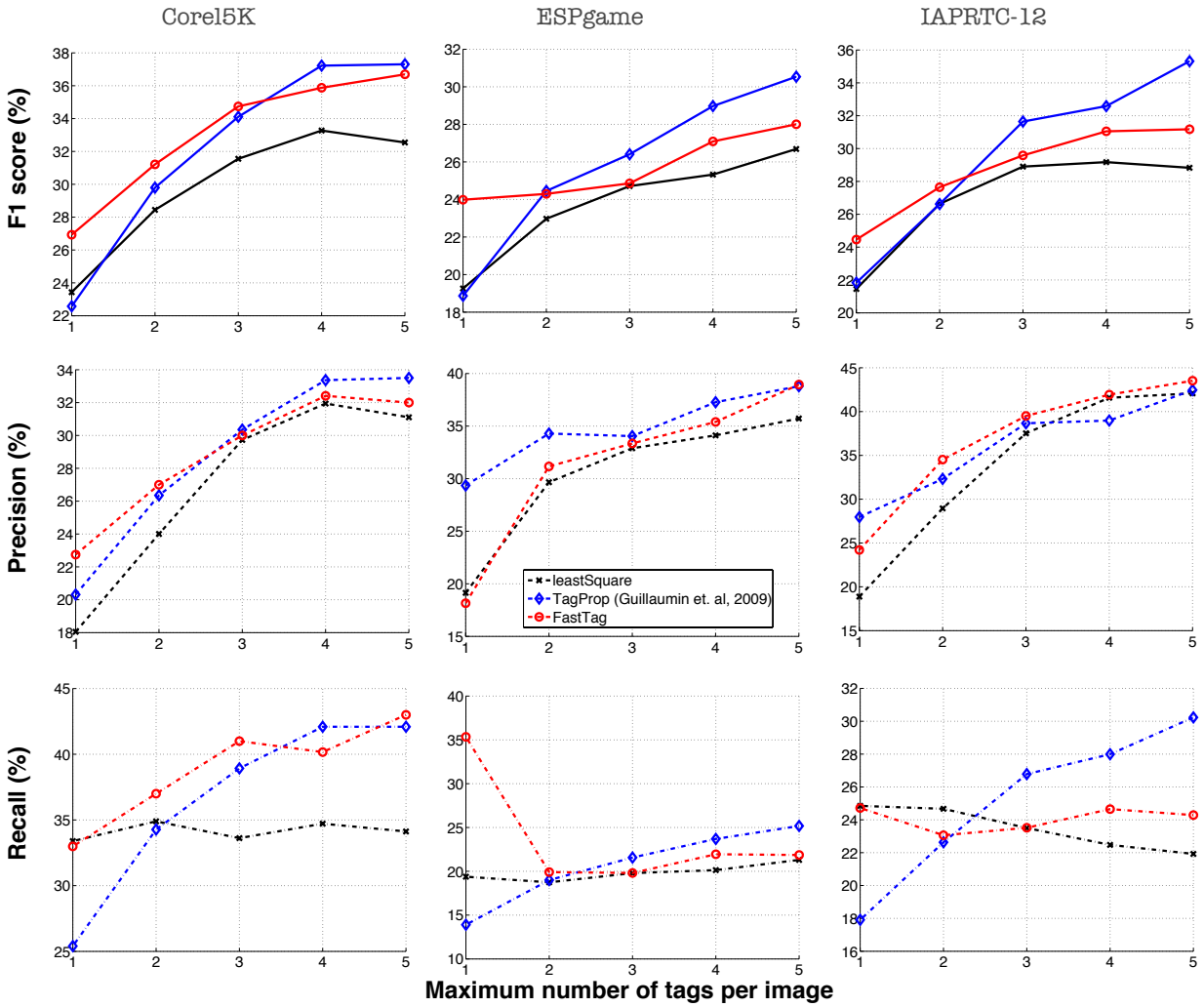

Figure 4. Performance in terms of Precision (P), Recall (R) and F1 score (F1) as a function of the maximum number of tags provided for each training image on the three benchmark datasets. The graphs compare the results of FastTag with the TagProp algorithm at different levels of tag sparsity.

increase the number of tags with positive recall by 34.

Table 2 compares the performance of FastTag over leastSquares and three existing methods on the ESP game and IAPRTC-12 datasets. Similar trends are observed. First, FastTag significantly outperforms the baseline, MBRM (a generative mixture model) of Feng et al. (2004), and JEC (a local NN method) of Makadia et al. (2008) on both datasets. FastTag performs slightly worse than TagProp. However, as we demonstrate next, FastTag achieves enormous speedup over TagProp in both training and testing.

Computational time. All experiments were conducted on a desktop with dual 6-core Intel i7 cpus with 2.66Ghz.

Figure 3 shows the F1 score vs. the training time required for different methods on these three datasets. The time is plotted in log scale. We can make three observations: 1. TagProp outperforms all other related work in terms of F1 measure, but is also the slowest to train. It takes close to one hour to train on the relatively small Corel5K dataset, which has around 4,500 training examples. For the larger datasets (ESPgame and IAPRTC-12) with close to 17,000 examples, the training time blows up to 16 hours. 2. The JEC method of (Makadia et al., 2008) falls into the same category of local NN method as TagProp, with the difference that it uses the simple average of the 15 distance metrics to define neighbors. JEC does not require training. However, we can see that it cannot compete in terms of accuracy performance. Note that, it still has $O(n)$ test-time complexity, where n is the number of training examples, because each query example requires a neighbor-lookup during testing. 3. The training time of FastTag is over 50x faster than that of TagProp. Note the time reported in the figure for FastTag also includes the feature preprocessing time, *i.e.*, performing homogeneous feature mapping and random projection, which takes up the majority of the computation time. For a total of 16,748 training examples (dimensionality $d = 15,000$) and 268 tags, FastTag takes on average 34 seconds to train for one bootstrap iteration. The optimal number of bootstrap iterations ranges from 1 to 8 in different re-optimization iterations (The number of iterations is usually very small at the beginning, but gradually increases in the later re-optimization stages as it needs bootstrapping to recover rare tags.). The algorithm converges within a few re-optimization stages.

4.3. Sample annotations

Figure 2 shows example images from the ESP game data set and their tag annotations obtained with FastTag. The figure shows three rows of results. The top row consists of images with high $F1$ score, *i.e.* these are images on which FastTag reliably retrieves relevant tags. The middle row shows images that are picked uniformly at random. Although not perfect, the vast majority of tags are relevant to the particular image. The bottom images have low $F1$ score, and represent examples where FastTag fails to retrieve relevant tags.

4.4. Further analysis

While these benchmark data sets are appropriate for algorithm comparisons, they may not be representative of the quality of training image tags found in the wild. In practice, most of the images are annotated with far fewer tags. We run the algorithms on images with down-sampled sparse tags in order to gauge their performance in this more realistic setting. Figure 4 depicts the comparison of FastTag and TagProp 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 (down sampled from the full set if more tags are available), then up to two tags, and so on. We can see that FastTag out-performs TagProp when the maximum number of provided tags is small. In general, FastTag performs comparably to Tagprop across different tag sparsity levels. In other words, the tag enrichment mapping of FastTag indeed helps to alleviate the intrinsic tag sparsity problem.

5. Conclusions

We present an image tagging method, FastTag, that performs on-par with current state-of-the-art algorithms, at a fraction of the computation cost. We recast a supervised multi-label classification problem as unlabeled multi-view learning. We define two classifiers, one for each view of the data, and coerce them into agreement via co-regularization in a joint loss function. We trade off complexity in the classifiers with non-linear mapping of the features and demonstrate that such a choice pays off. FastTag is computationally efficient during training and testing yet maintains tagging accuracy. It can effectively deal with sparsely tagged training data and rare tags that are often obstacles in such large-scale learning problems.

Acknowledgments

We thank David Grangier and Larry Zitnick of Microsoft research for helpful discussions. KQW was supported by NSF grants 1149882 and 1137211. Part of this work was done while MC was an intern at Microsoft Research, Redmond.

References

Barnard, Kobus, Duygulu, Pinar, Forsyth, David, De Freitas, Nando, Blei, David M, and Jordan, Michael I. Matching words and pictures. *The Journal of Machine Learning Research*, 3:1107–1135, 2003.

Carneiro, G., Chan, A.B., Moreno, P.J., and Vasconcelos, N. Supervised learning of semantic classes for image annotation and retrieval. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(3):394–410, 2007.

Chen, M., Xu, Z., Weinberger, K., and Sha, F. Marginalized denoising autoencoders for domain adaptation. In *ICML '12*, pp. 767–774. ACM, New York, NY, USA, July 2012.

Cusano, Claudio, Ciocca, Gianluigi, and Schettini, Raimondo. Image annotation using svm. In *Electronic Imaging 2004*, pp. 330–338. International Society for Optics and Photonics, 2003.

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

Feng, SL, Manmatha, R., and Lavrenko, V. Multiple bernoulli relevance models for image and video annotation. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pp. II–1002. IEEE, 2004.

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

Grangier, David and Bengio, Samy. A discriminative kernel-based approach to rank images from text queries. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(8):1371–1384, 2008.

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

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

Hafner, J., Sawhney, H.S., Equitz, W., Flickner, M., and Niblack, W. Efficient color histogram indexing for quadratic form distance functions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(7):729–736, 1995.

Hertz, Tomer, Bar-Hillel, Aharon, and Weinshall, Daphna. Learning distance functions for image retrieval. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pp. II–570. IEEE, 2004.

Jeon, J., Lavrenko, V., and Manmatha, R. 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 information retrieval*, pp. 119–126. ACM, 2003.

Lavrenko, V., Manmatha, R., and Jeon, J. A model for learning the semantics of pictures. NIPS, 2003.

Liu, J., Li, M., Liu, Q., Lu, H., and Ma, S. Image annotation via graph learning. *Pattern Recognition*, 42(2): 218–228, 2009.

Lowe, D.G. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pp. 1150–1157. Ieee, 1999.

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

Metzler, D. and Manmatha, R. An inference network approach to image retrieval. *Image and video retrieval*, pp. 2130–2131, 2004.

Monay, Florent and Gatica-Perez, Daniel. Plsa-based image auto-annotation: constraining the latent space. In *Proceedings of the 12th annual ACM international conference on Multimedia*, pp. 348–351. ACM, 2004.

Oliva, A. and Torralba, A. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

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

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

Vedaldi, A. and Zisserman, A. Efficient additive kernels via explicit feature maps. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(3):480–492, 2012.

Vempala, Santosh S. *The random projection method*, volume 65. Amer Mathematical Society, 2005.

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

Weinberger, Kilian Q., Slaney, Malcolm, and Van Zwol, Roelof. Resolving tag ambiguity. In *Proceeding of the 16th ACM international conference on Multimedia*, MM '08, pp. 111–120, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-303-7.

Yavlinsky, A., Schofield, E., and Rüger, S. Automated image annotation using global features and robust nonparametric density estimation. *Image and video retrieval*, pp. 593–593, 2005.