Learning from Ambiguously Labeled Images

Timothee Cour [*] Benjamin Sapp [†]
Chris Jordan [‡] Ben Taskar [**]

[*]University of Pennsylvania,
[†]University of Pennsylvania,
[‡]University of Pennsylvania,
[**]University of Pennsylvania, taskar@seas.upenn.edu

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-09-07

Learning from Ambiguously Labeled Images

Timothee Cour, Benjamin Sapp, Chris Jordan, Ben Taskar
University of Pennsylvania
3330, Walnut Street
{timothee,bensapp,wjc,taskar}@seas.upenn.edu

Abstract

In many image and video collections, we have access only to partially labeled data. For example, personal photo collections often contain several faces per image and a caption that only specifies who is in the picture, but not which name matches which face. Similarly, movie screenplays can tell us who is in the scene, but not when and where they are on the screen. We formulate the learning problem in this setting as partially-supervised multiclass classification where each instance is labeled ambiguously with more than one label. We show theoretically that effective learning is possible under reasonable assumptions even when all the data is weakly labeled. Motivated by the analysis, we propose a general convex learning formulation based on minimization of a surrogate loss appropriate for the ambiguous label setting. We apply our framework to identifying faces culled from web news sources and to naming characters in TV series and movies. We experiment on a very large dataset consisting of 100 hours of video, and in particular achieve 6% *error for character naming on 16 episodes of LOST.*

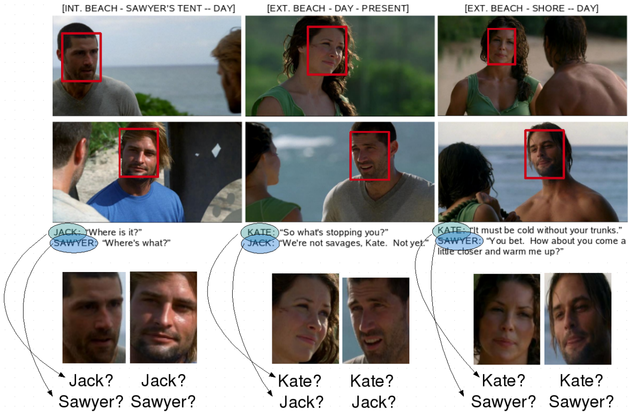
[INT. BEACH - SAWYER'S TENT -- DAY]
[EXT. BEACH - DAY - PRESENT]
[EXT. BEACH - SHORE -- DAY]
JACK: "Where is it?"
SAWYER: "Where's what?"
KATE: "So what's stopping you?"
JACK: "We're not savages, Kate. Not yet."
KATE: "It must be cold without your trunks."
SAWYER: "You bet. How about you come a little closer and warm me up?"
Jack? Sawyer?
Jack? Sawyer?
Kate? Jack?
Kate? Jack?
Kate? Sawyer?
Kate? Sawyer?

Figure 1. Examples of frames and corresponding parts of the script from the TV series "LOST". From aligning the script to the video, we have 2 ambiguous labels for each person in the 3 different scenes.

1. Introduction

Photograph collections with captions have motivated recent interest in weakly annotated images [5, 1]. As a further motivation, consider Figure 1, which shows another common setting where we can obtain plentiful but ambiguously labeled data: videos and screenplays. Using a screenplay, we can tell who is in the scene, but for every face in the images, the person's identity is ambiguous. Learning accurate face and object recognition models from such imprecisely annotated images and videos can improve many applications, including image retrieval and summarization. In this paper, we investigate theoretically and empirically when effective learning from this weak supervision is possible.

To put the ambiguous labels learning problem into perspective, it is useful to lay out several related learning scenarios. In **semi-supervised** learning, the learner has access to a set of labeled examples as well as a set of unlabeled examples. In **multiple-instance** learning, examples are not individually labeled but grouped into sets which either contain at least 1 positive example, or only negative examples. In **multi-label** learning, each example is assigned multiple binary labels, all of which can be true. Finally, in our setting of **ambiguous labeling**, each example again is supplied with multiple potential labels, *only one of which is correct.* A formal definition is given in Sec. 3.

There have been several papers that addressed the ambiguous label framework. [11] proposes several nonparametric, instance-based algorithms for ambiguous learning based on greedy heuristics. [12] uses expectation-maximization (EM) algorithm with a discriminative log-linear model to disambiguate correct labels from incorrect. Additionally, these papers only report results on synthetically-created ambiguous labels and rely on iterative non-convex optimization.

In this work, we provide intuitive assumptions under which we can expect learning to succeed. Essentially, we identify a condition under which ambiguously labeled data is sufficient to compute a useful upper bound on the true labeled error. We propose a simple, convex formulation based on this analysis and show how to extend general multi-class loss functions to handle ambiguity. We show that our method significantly outperforms several strong baselines on a large dataset of pictures from newswire and a large video collection.

2. Related work

A more general multi-class setting is common for images with captions (for example, a photograph of a beach with a palm and a boat, where object locations are not specified). [5, 1] show that such partial supervision can be sufficient to learn to identify the object locations. The key observation is that while text and images are separately ambiguous, jointly they complement each other. The text, for instance, does not mention obvious appearance properties, but the frequent co-occurrence of a word with a visual element could be an indication of association between the word and a region in the image. Of course, words in the text without correspondences in the image and parts of the image not described in the text are virtually inevitable. The problem of naming image regions can be posed as translation from one language to another. Barnard et al. [1] address it using a multi-modal extension to mixture of latent Dirichlet allocation.

The specific problem of naming faces in images and videos using text sources has been addressed in several works [14, 2, 8, 6]. There is vast literature on fully supervised face recognition, which is out of the scope of this thesis. Approaches relevant to ours include [2] which aims at clustering face images obtained by detecting faces from images with captions. Since the name of the depicted people typically appears in the caption, the resulting set of images is ambiguously labeled, if more than one name appears in the caption. Moreover, in some cases the correct name may not be included in the set of potential labels for a face. The problem can be solved by using unambiguous images to estimate discriminant coordinates for the entire dataset. The images are clustered in this space and the process is iterated. Gallagher and Chen [8] address the similar problem of retrieval from consumer photo collections, in which several people appear in each image which is labeled with their names. Instead of estimating a prior probability for each individual, the algorithm estimates a prior for groups using the ambiguous labels. Unlike [2], the method of [8] does not handle erroneous names in the captions.

In work on video, a wide range of cues was used to help supervise the data, including: using captions or transcripts [6], using sound [14] to obtain the transcript, using clustering based on clothing within scenes to group instances [13]. Most of the methods involve either procedural, iterative reassignment schemes or non-convex optimization.

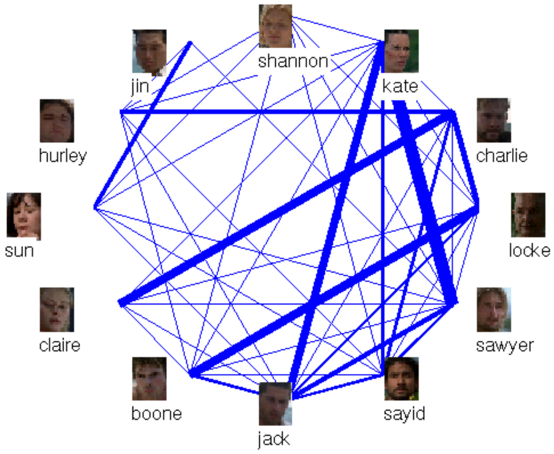

Figure 2. Co-occurrence graph of the top characters across 16 episodes of LOST. Larger edges correspond to a pair of characters appearing together more frequently in the season.

3. Formulation

In the standard supervised multiclass setting, we have labeled examples $S = \{(x_i, y_i)_{i=1}^m\}$ from an unknown distribution $P(x, y)$ where $x \in \mathcal{X}$ is the input and $y \in \{1, \ldots, L\}$ is the class label. In the partially supervised setting we investigate, instead of an unambiguous single label per instance we have a set of labels, one of which is the correct label for the instance. We will denote the sample as $S = \{(x_i, y_i, Z_i)_{i=1}^m\}$ from an unknown distribution $P(x, y, Z) = P(x, y)P(Z \mid x, y)$ where $Z_i \subseteq \{1, \ldots, L\} \setminus y_i$ is a set of additional labels. We will denote $Y_i = y_i \cup Z_i$ as the ambiguity set actually observed by the learning algorithm. Clearly, our setup generalizes the standard semi-supervised setting where some examples are labeled and some are unlabeled: if the ambiguity set Y_i includes all the labels, the example is unlabeled and if the ambiguity set contains one label, we have a labeled example. We consider the middle-ground, where all examples are partially labeled as described in our motivating examples and analyze assumptions under which learning can be guaranteed to succeed.

Consider a very simple ambiguity pattern that makes learning impossible: $L = 3$, $|Z_i| = 1$ and label 1 is present in every set Y_i. Then we cannot distinguish between the case where 1 is the true label of every example or the case where it is not a label of any example. More generally, if two labels always co-occur when present in Y, we cannot tell them apart. In order to learn from ambiguous data, we need to make some assumptions about the joint distribution of $P(Z \mid x, y)$. Below we will make an assumption that ensures some diversity in the ambiguity set. Looking at Figure 2, we can see that the distribution of ambiguous pairs is more benign.

The model and loss functions. We assume a mapping $\mathbf{f}(x) : \mathcal{X} \mapsto \Re^d$ from inputs to d real-valued features and a multi-linear classifier $g(x) : \mathcal{X} \mapsto \Re^L$ with L components,

$$g^a(x) = \mathbf{w}^a \cdot \mathbf{f}(x),$$

one for each label $a \in \{1, \ldots, L\}$, to which we will refer to as class scores. The prediction of the classifier is determined by $g^*(x) = \arg\max_a g^a(x)$, the highest scoring label according to g^a (we assume that ties are broken arbitrarily, for example, by selecting the label with smallest index a). Hence the classifier is parameterized by $d \times L$ weights w_i^a, one for each feature-and-class pair.

Many formulations of fully-supervised multiclass learning have been proposed based on minimization of convex upper bounds on risk, usually, the $0/1$ loss [16]:

$$\mathcal{L}_{01}(g(x), y) = \mathbb{1}(g^*(x) \neq y).$$

In addition, we define ambiguous $0/1$ loss:

$$\mathcal{L}_{01}(g(x), Y) = \mathbb{1}(g^*(x) \notin Y).$$

Connection between ambiguous and standard $0/1$ loss. An obvious observation is that the ambiguous loss is an underestimate of the true loss. However in the ambiguous learning setting we would like to minimize the $0/1$, with access only to the ambiguous loss. Therefore we need a way to upperbound the $0/1$ loss with the ambiguous loss.

The following definition defines a measure of the hardness of learning under ambiguous supervision.

Definition. Ambiguity degree $\epsilon(P)$ of a distribution We define the ambiguity degree $\epsilon(P)$ of a distribution $P(x, y, Z)$ as:

$$\epsilon(P) = \sup_{x \in \mathcal{X}; y, a \in \{1,\ldots,L\}} P(a \in Z \mid x, y). \tag{1}$$

In words, $\epsilon(P)$ corresponds to the maximum probability of an extra label co-occurring with a true label y, over all labels and examples. Let us consider several extreme cases: When $\epsilon(P) = 0$, $Z = \emptyset$ with probability one, and we are back to standard supervised learning case, with no ambiguity. When $\epsilon(P) = 1$, some extra label consistently co-occurs with a true label y on an example x and we cannot tell them apart: no learning is possible for this example. For a fixed ambiguity set size $|Z|$, the smallest possible ambiguity degree is achieved for the uniform case: $\epsilon(P) = |Z|/(L-1)$. Intuitively, the best case scenario for ambiguous learning corresponds to a distribution with high conditional entropy for $P(Z|x, y)$.

The following proposition shows we can bound the (unobserved) $0/1$ loss by the (observed) ambiguous loss, allowing us to approximately minimize the standard loss with only access to the ambiguous one. The tightness of the approximation directly relates to the ambiguity degree.

Proposition 3.1 *For any classifier g and distribution P with $\epsilon(P) < 1$,*

$$\mathbf{E}_P[\mathcal{L}_{01}(g(x), Y)] \leq \mathbf{E}_P[\mathcal{L}_{01}(g(x), y)] \tag{2}$$

$$\leq \frac{1}{1-\epsilon(P)} \mathbf{E}_P[\mathcal{L}_{01}(g(x), Y)] \tag{3}$$

Proof The first inequality comes from the fact that $g^*(x) \notin Y \implies g^*(x) \neq y$. For the second inequality, fix an $x \in \mathcal{X}$ and define $\mathbf{E}_P[\cdot \mid x]$ as the expectation with respect to $P(Y \mid x) = P(y, Z \mid x)$.

$$\begin{aligned}
&\mathbf{E}_P[\mathcal{L}_{01}(g(x), Y)|x] = P(g^*(x) \notin Y \mid x) \\
&= P(y \neq g^*(x), g^*(x) \notin Z \mid x) \\
&= \sum_{a \neq g^*(x)} P(y = a \mid x) \underbrace{(1 - P(g^*(x) \in Z \mid x, y = a))}_{\geq 1 - \epsilon(P)} \\
&\geq \sum_{a \neq g^*(x)} P(y = a \mid x)(1 - \epsilon(P)) \\
&= (1 - \epsilon(P)) \mathbf{E}_P[\mathcal{L}_{01}(g(x), y)|x]
\end{aligned}$$

Hence, $\mathbf{E}_P[\mathcal{L}_{01}(g(x), y)|x] \leq \frac{1}{1-\epsilon(P)} \mathbf{E}_P[\mathcal{L}_{01}(g(x), Y)|x]$ for any x. We conclude by taking expectation over x □

Note, the second bound is tight, as can be shown by considering the uniform case with a fixed ambiguity size Z and $P(a \in Z \mid x, y) = |Z|/(L-1)$.

Robustness to outliers. One potential issue with proposition 3.1 is that unlikely pairs x, y might force ϵ to be large, making the bound very loose. We show we can refine the notion of ambiguity degree $\epsilon(P)$ by excluding such pairs.

Definition. (ϵ, δ)-ambiguous distribution. Define a distribution P to be (ϵ, δ)-ambiguous if there is a subset of the space $A \subseteq \mathcal{X} \times \{1, \ldots, L\}$ with probability mass at least $1 - \delta$, (i.e. $P((x, y) \in A) \geq 1 - \delta$), where

$$\sup_{(x,y) \in A, a \in \{1,\ldots,L\}} P(a \in Z \mid x, y) \leq \epsilon$$

Note, in the extreme case $\epsilon = 0$, this corresponds to standard semi-supervised learning, where δ-proportion of examples are unambiguously labeled, and $1 - \delta$ are (potentially) fully unlabeled.

This definition allows us to bound the $0/1$ loss even in the case when some unlikely pair x, y with probability $\leq \delta$ would make the ambiguity degree arbitrarily large. Suppose we mix an initial distribution with small ambiguity degree, with an outlier distribution with large overall ambiguity degree. The following proposition shows that the bound degrades only by an additive amount, which can be interpreted as a form of robustness to outliers.

Proposition 3.2 *For any classifier g and (ϵ, δ)-ambiguous $P(Z \mid x, y)$,*

$$\mathbf{E}_P[\mathcal{L}_{01}(g(x), y)] \leq \frac{1}{1-\epsilon} \mathbf{E}_P[\mathcal{L}_{01}(g(x), Y)] + \delta.$$

Proof We split up the expectation in two parts:

$$\begin{aligned}
&\mathbf{E}_P[\mathcal{L}_{01}(g(x),y)] = \mathbf{E}_P[\mathcal{L}_{01}(g(x),y)|(x,y) \in A](1-\delta)\\
&+ \mathbf{E}_P[\mathcal{L}_{01}(g(x),y)|(x,y) \notin A]\delta\\
&\leq \mathbf{E}_P[\mathcal{L}_{01}(g(x),y)|(x,y) \in A](1-\delta) + \delta\\
&\leq \frac{1}{1-\epsilon}\mathbf{E}_P[\mathcal{L}_{01}(g(x),Y)|(x,y) \in A](1-\delta) + \delta
\end{aligned}$$

We applied proposition 3.1 in the last step. Using a symmetric argument,

$$\begin{aligned}
\mathbf{E}_P[\mathcal{L}_{01}(g(x),Y)] &= \mathbf{E}_P[\mathcal{L}_{01}(g(x),Y)|(x,y) \in A](1-\delta)\\
&+ \mathbf{E}_P[\mathcal{L}_{01}(g(x),Y)|(x,y) \notin A]\delta\\
&\geq \mathbf{E}_P[\mathcal{L}_{01}(g(x),Y)|(x,y) \in A](1-\delta)
\end{aligned}$$

Finally we obtain $\mathbf{E}_P[\mathcal{L}_{01}(g(x),y)] \leq \frac{1}{1-\epsilon}\mathbf{E}_P[\mathcal{L}_{01}(g(x),Y)] + \delta$ □

Label-specific recall bounds. In real settings such as in our movie experiments we observe that certain subsets of labels are harder to disambiguate than others. We can further tighten our bounds between ambiguous loss and standard $0/1$ loss if we consider label specific information. We define the *label-specific ambiguity degree* $\epsilon^a(P)$ of a distribution (with $a \in \{1,\dots,L\}$) as:

$$\epsilon^a(P) = \sup_{x \in \mathcal{X}; a' \in \{1,\dots,L\}} P(a' \in Z \mid x, y=a).$$

We can show a label-specific analog of proposition 3.1:

Proposition 3.3 *For any classifier g and distribution P with $\epsilon^a(P) < 1$,*

$$\mathbf{E}_P[\mathcal{L}_{01}(g(x),y) \mid y=a] \leq \frac{1}{1-\epsilon^a}\mathbf{E}_P[\mathcal{L}_{01}(g(x),Y) \mid y=a],$$

where we see that ϵ^a bounds per-class recall.

Proof Fix an $x \in \mathcal{X}$ (such that $P(y=a|x) > 0$) and define $\mathbf{E}_P[\cdot \mid x, y=a]$ as the expectation with respect to $P(Z \mid x, y=a)$. We consider two cases:

a) if $g^*(x) = a$,

$$\begin{aligned}
&\mathbf{E}_P[\mathcal{L}_{01}(g(x),Y) \mid x, y=a]\\
&= P(g^*(x) \neq y, g^*(x) \notin Z \mid x, y=a) = 0
\end{aligned}$$

b) if $g^*(x) \neq a$,

$$\begin{aligned}
&\mathbf{E}_P[\mathcal{L}_{01}(g(x),Y) \mid x, y=a]\\
&= P(g^*(x) \notin Z \mid x, y=a)\\
&= 1 - P(g^*(x) \in Z \mid x, y=a) \geq 1-\epsilon^a
\end{aligned}$$

We conclude by taking expectation over x:

$$\begin{aligned}
&\mathbf{E}_P[\mathcal{L}_{01}(g(x),Y) \mid y=a]\\
&= P(g^*(x)=a|y=a)\mathbf{E}_P[\mathcal{L}_{01}(g(x),Y) \mid g^*(x)=a, y=a]\\
&+ P(g^*(x)\neq a|y=a)\mathbf{E}_P[\mathcal{L}_{01}(g(x),Y) \mid g^*(x)\neq a, y=a]\\
&\geq 0 + P(g^*(x) \neq a \mid y=a) \cdot (1-\epsilon^a)\\
&= (1-\epsilon^a) \cdot \mathbf{E}_P[\mathcal{L}_{01}(g(x),y) \mid y=a] \square
\end{aligned}$$

These bounds give a strong give a strong connection between ambiguous loss and real loss, which allows us to approximately minimize the expected real loss by minimizing (an upper bound on) the ambiguous loss.

4. A convex learning formulation

We build our formulation on a simple and general multiclass scheme that combines convex binary losses $\psi(\cdot) : \Re \mapsto \Re_+$ on individual components of g to create a multiclass loss. For example, we can use hinge, exponential or logistic loss. In particular, we assume a type of one-against-all scheme for the supervised case:

$$\mathcal{L}_\psi(g(x),y) = \psi(g^y(x)) + \sum_{a \neq y} \psi(-g^a(x)). \qquad (4)$$

A classifier g is selected by minimizing the empirical loss on the sample augmented with a regularization term to penalize complex models.

Convex loss for ambiguous labels. In the ambiguous label setting, instead of an unambiguous single label y per instance we have a set of labels Y, one of which is the correct label for the instance. We propose the following loss function:

$$\boxed{\mathcal{L}_\psi(g(x),Y) = \psi\left(\frac{1}{|Y|}\sum_{a \in Y} g^a(x)\right) + \sum_{a \notin Y} \psi(-g^a(x)) \quad (5)}$$

Note that if the set Y contains a single label y, then the loss function reduces to the regular multiclass loss. When Y is not a singleton, then the loss function will drive up the *average* of the scores of the labels in Y. If the score of the correct label is large enough, the other labels in the set do not need to be positive. This tendency alone does not guarantee that the correct label has the *highest* score. However, we show in (8) that $\mathcal{L}_\psi(g(x),Y)$ upperbounds $\mathcal{L}_{01}(g(x),Y)$ whenever $\psi(\cdot)$ is an upper bound on the $0/1$ loss.

Of course, minimizing an upperbound on the loss does not always lead to sensible algorithms. We show next that our convex relaxation offers a tighter upperbound to the ambiguous loss compared to a more straightforward multi-label approach.

Comparison to naive multi-label loss. The "naive" model treats each example as taking on multiple correct labels, which implies the following loss function

$$\mathcal{L}_\psi^{naive}(g(x), Y) = \sum_{a \in Y} \psi\left(g^a(x)\right) + \sum_{a \notin Y} \psi(-g^a(x)) \quad (6)$$

One reason we expect our loss function to outperform the naive approach is that we obtain a tighter convex upper bound on $\mathcal{L}_{01}$. Let us also define

$$\mathcal{L}_\psi^{max}(g(x), Y) = \psi\left(\max_{a \in Y} g^a(x)\right) + \sum_{a \notin Y} \psi(-g^a(x)) \quad (7)$$

which is not convex. Under the usual conditions that ψ is a convex, decreasing upper bound of the step function (e.g., square hinge loss, exponential loss, and log loss with proper scaling), the following inequalities hold:

Proposition 4.1 (comparison between ambiguous losses)

$$\mathcal{L}_{01} \leq \mathcal{L}_\psi^{max} \leq \mathcal{L}_\psi \leq \mathcal{L}_\psi^{naive} \quad (8)$$

Proof For the first inequality, if $g^*(x) \in Y$, $\mathcal{L}_\psi^{max}(g(x), Y) \geq 0 = \mathcal{L}_{01}(g(x), Y)$. Otherwise we have two cases with $a^* = g^*(x) \notin Y$:

a) if $g^{a^*}(x) \leq 0$, $\max_{a \in Y} g^a(x) \leq 0$ by definition of $g^*(x)$ so $\psi(\max_{a \in Y} g^a(x)) \geq 1$

b) if $g^{a^*}(x) > 0$, $\psi(-g^{a^*}(x)) \geq 1$

In both cases, $\mathcal{L}_\psi^{max}(g(x), Y) \geq 1 = \mathcal{L}_{01}(g(x), Y)$.
The second inequality comes from the fact that

$$\max_{a \in Y} g^a(x) \geq \frac{1}{|Y|} \sum_{a \in Y} g^a(x)$$

For the third inequality, using the convexity of ψ,

$$\psi\left(\frac{1}{|Y|} \sum_{a \in Y} g^a(x)\right) \leq \frac{1}{|Y|} \sum_{a \in Y} \psi(g^a(x)) \leq \sum_{a \in Y} \psi(g^a(x)) \quad \square$$

This shows that our loss $\mathcal{L}_\psi$ is a tighter approximation to $\mathcal{L}_{01}$ than $\mathcal{L}_\psi^{naive}$, as illustrated in figures 3 and 4. What's more, the bound is non-trivial: when $g^a(x) = constant$ over $a \in Y$, we have

$$\psi\left(\max_{a \in Y} g^a(x)\right) = \psi\left(\frac{1}{|Y|} \sum_{a \in Y} g^a(x)\right) = \frac{1}{|Y|} \sum_{a \in Y} \psi\left(g^a(x)\right)$$

To gain additional intuition on why our proposed loss (5) is better than the naive loss (6): For an input x with ambiguous label set (a, b), our model only encourages the *sum* $g^a(x) + g^b(x)$ to be large, allowing the correct score to be positive and the extraneous score to be negative (e.g., $g^a(x) = 2, g^b(x) = -1$). In contrast, the naive model encourages both $g^a(x)$ and $g^b(x)$ to be large.

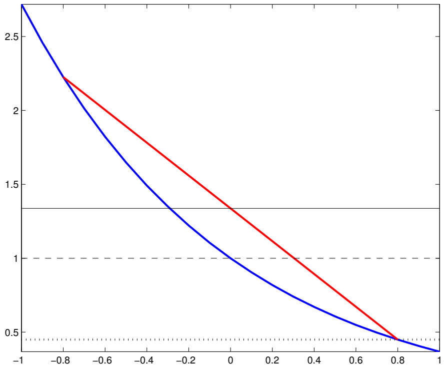

Figure 3. For a strictly convex loss such as the exp-loss (blue curve), the ambiguous loss provides a better approximation to the max-loss than the naive loss. The red segment corresponds to the chord with points g^1, g^2, the dashed line corresponds to $\psi(\frac{1}{2}(g^1 + g^2))$, the dotted line corresponds to $\psi(\max(g^1, g^2))$, and the black line corresponds to $\frac{1}{2}(\psi(g^1) + \psi(g^2))$.

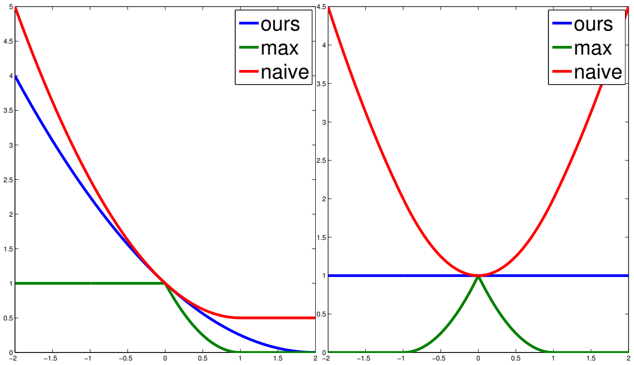

Figure 4. Our loss, (5) provides a tighter upperbound than the naive loss (6) on the non-convex function (7). **Left:** plots of $\psi(\frac{1}{2}(g^1 + g^2))$ (ours), $\psi(\max(g^1, g^2))$ (max), $\frac{1}{2}(\psi(g^1) + \psi(g^2))$ (naive), as a function of $g^1 \in [-2, 2]$ (with $g^2 = 0$ fixed). **Right:** same, with $g^2 = -g^1$. In each case we the square hinge loss for ψ, assume $Y = \{1, 2\}$, and drop the negative terms.

Algorithm. Our ambiguous learning formulation is flexible and we can derive many alternative algorithms depending on the choice of the binary loss $\psi(u)$, the regularization, and the optimization method. In the experiments we use the square hinge loss for ψ and add an L_2 regularization, resulting in the following objective:

$$\min_{\mathbf{w}} \frac{1}{2}||\mathbf{w}||_2^2 + C||\xi||_2^2 \tag{9}$$

$$\text{s.t.} \quad \frac{1}{|Y_i|} \sum_{a \in Y_i} \mathbf{w}^a \cdot \mathbf{f}(x_i) \geq 1 - \xi_i \tag{10}$$

$$-\mathbf{w}^a \cdot \mathbf{f}(x_i) \geq 1 - \xi_{ia} \quad \forall a \notin Y_i \tag{11}$$

where $\{\xi_i, \xi_{ia}\}$ are slack variables and C is a regularization parameter that can be set by K-fold cross-validation on the ambiguously labeled data. We fixed $C = 10^3$ in all experiments. The optimization can be converted into a L_2 loss linear Support Vector Machine, which we solve in the primal using a trust region Newton method, with the off-the-shelf implementation of [7]. The sparse structure of the problem allows us to tackle large scale problems with thousands of instances and features, and hundreds of labels.

5. Controlled experiments

We first perform a series of controlled experiments to analyze our algorithm on a face naming task from Labeled Faces in the Wild [10]. The goal is to correctly label faces from examples that have multiple potential labels (transductive case), as well as learn a model from ambiguous data that generalizes to other unlabeled examples (inductive case).

5.1. Baselines

In the experiments, we compare our approach with the following baselines.

Random model. We define *chance* as randomly guessing between the possible ambiguous labels only. Defining the (empirical) average ambiguous size to be $E[|Y|] = \frac{1}{m}\sum_{i=1}^{m} |Y_i|$, then the error from the *chance* baseline is given by $\text{error}_{\text{chance}} = 1 - \frac{1}{E[|Y|]}$.

IBM Model 1. This generative model was originally proposed in [3] for machine translation, but we can adapt it to the ambiguous label case. In our setting, the conditional probability of an example $x \in \Re^d$ belonging to one of its ambiguous labels $a \in Y$ is normally distributed. We use the expectation-maximization (EM) algorithm to learn the parameters of the Gaussians (mean μ_a and diagonal covariance matrix $\Sigma_a = diag(\sigma_a)$ for each label).

Discriminative EM. We compare with the model proposed in [12], which is a discriminative model with an EM procedure adapted for the ambiguous label setting. The authors minimize the KL divergence between a maximum entropy model P (estimated in the M-step) and a distribution over ambiguous labels $\hat{P}$ (estimated in the E-step):

$$J(\theta, \hat{P}) = \sum_i \sum_{y \in Y} \hat{P}(y|x_i) \log \left(\frac{\hat{P}(y|x_i)}{P(y|x_i, \theta)} \right) \tag{12}$$

k-Nearest Neighbor. Following [11], we adapt the k-Nearest Neighbor Classifier to the ambiguous label setting as follows:

$$g_k(x) = \arg\max_{y \in Y} \sum_{i=1}^{k} w_i \mathbb{1}(y \in Y_i) \tag{13}$$

where x_i is the i^{th} nearest-neighbor of x using Euclidean distance, and w_i are a set of weights. We use two kNN baselines: **kNN** assumes uniform weights $w_i = 1$ (model used in [11]), and **weighted kNN** uses linearly decreasing weights $w_i = k - i + 1$. We use $k = 5$ and break ties randomly as in [11].

Naive model. This is introduced in (6). After training, we predict the label with the highest score (in the transductive setting): $y = \arg\max_{a \in Y} g^a(x)$.

Supervised models. Finally we also consider two baselines that *ignore* the ambiguous label setting. The first one, denoted as **supervised model**, removes from (6) the examples with $|Y| > 1$. The second model, denoted as **supervised kNN**, removes from (13) the same examples.

5.2. Faces in the Wild

We experiment with a subset of the publicly available Labeled Faces in the Wild [9] dataset. We take the first 50 images of the top 10 most frequent people, yielding a balanced dataset for controlled experiments.

Features. We use the images registered with funneling, and crop out the central part corresponding to the approximate face location, which we resize to 60x90. We project the resulting grayscale patches (treated as 5400x1 vectors) onto a 50-dimensional subspace using PCA[1].

Experimental setup. For the **inductive experiments**, we split randomly in half the instances into (1) **ambiguously labeled training set**, and (2) **unlabeled testing set**. The ambiguous labels in the training set are generated randomly according to different noise models which we specify in each case. For each method and parameter setting, we report the **average test error rate** over **20 trials** after training the model on the ambiguous train set. We also report the corresponding **standard deviation** as error bar in the plots. Note, in the inductive setting we consider the test set is unlabeled, and so the classifier votes among *all* possible labels:

$$y = \arg\max_{a \in \{1..L\}} g^a(x) \tag{14}$$

For the **transductive experiments**, there is no test set; we report the error rate for disambiguating the ambiguous labels (also averaged over 20 trials corresponding to random

[1] We kept the features simple by design; more sophisticated part-based registration and representation and would further improve results, as we will see in section 6

settings of ambiguous labels). The main differences with the inductive setting are: (1) the model is trained on all instances and tested on the same instances; and (2) the classifier votes only among the ambiguous labels, which is easier:

$$y = \arg\max_{a \in Y} g^a(x) \tag{15}$$

We compare our approach (denoted as **mean**) against the **baselines** presented in section 5.1: Chance, Model 1, Discriminative EM model, k-Nearest Neighbor, weighted k-Nearest Neighbor, Naive model, supervised model, and supervised kNN. Note, in our experiments the Discriminative EM model was much slower to converge than all the other methods, and we only report the first series of experiments with this baseline.

In figure 5, we vary the **ambiguity size:** the number of extra labels associated with each example, normalized by the total number of labels to lie in the range [0,1]. In this setting, the ambiguous labels in the training set are generated uniformly without replacement. We plot the results in the inductive case for three different subsets of Faces in the Wild: a balanced dataset using 50 faces for each of the top 10 labels, an unbalanced dataset using all faces for each of the top 10 labels. and an unbalanced dataset using up to 100 faces for each of the top 100 labels.

In figure 6 we report additional results on the first dataset. On the left, we vary the ambiguity size in the transductive setting. In the middle plot we vary the **ambiguity degree** ϵ (defined in (1)) in the range [0,1], using the inductive setting. This is achieved by first choosing at random for each label a dominant co-occurring label which is sampled with probability ϵ; the rest of the labels are sampled uniformly with probability $(1-\epsilon)/(L-2)$ (there is a single extra label per example). Finally, in the right plot we vary the dimensionality (in the inductive setting), by increasing the number o PCA components from 1 to 200, with half of extra labels added uniformly at random.

There are several clear trends in Figure 5. Our method dominates in all settings, followed by the *naive* model. As is expected, increasing ambiguity size monotonously affects error rate. We also see that increasing ϵ significantly affects error, even though the ambiguity size is constant, consistent with our bounds in Section 3.

6. Ambiguously Labeled Faces on TV

We now return to our introductory motivating example, naming people in TV shows (Figure 1). Our goal is to identify characters given ambiguous labels derived from the screenplay. Our data consists of 100 episodes ($\sim$ 75 hours) of LOST and CSI, from which we extract ambiguously labeled faces to learn models of common characters. We use the same features, learning algorithm and loss function as in section 5.2. We also explore using additional person- and movie-specific constraints to improve performance.

Data Collection. We adopt the following filtering pipeline to extract face tracks, inspired by [6]:

(1) Run the off-the-shelf OpenCV face detector over all frames, searching over rotations and scales. **(2)** Run face part detectors[2] over the face candidates. **(3)** Perform a 2D rigid transform of the parts to a template. **(4)** Compute the score of a candidate face $s(x)$ as the sum of part detector scores plus rigid fit error, normalizing each to weight them equally, and filtering out faces with low score. **(5)** Assign faces to tracks by associating face detections within a shot using normalized cross-correlation in RGB space, and using dynamic programming to group them together into tracks. **(6)** Subsample face tracks to avoid repetitive examples. In the experiments reported here we use the best scoring face in each track, according to $s(x)$.

Concretely, for a particular episode, step (1) finds approximately 100,000 faces, step (4) keeps approximately 10,000 of those, and after subsampling tracks in step (6) we are left with 1000 face detections.

Ambiguous Label Selection. Screenplays for popular TV series and movies are readily available for free on the web. Given an alignment of the screenplay to frames, we have ambiguous labels for characters in each scene: the set of speakers mentioned at some point in the scene, as shown in Figure 1. Alignment of screenplay to video uses methods presented in [4, 6], linking closed captions to screenplay.

We use the ambiguous sets to select face tracks filtered through our pipeline. We prune scenes which contain characters other than the set we choose to focus on for experiments (top $\{8,16,32\}$ characters), or contain 4 or more characters. This leaves ambiguous bags of size 1, 2 or 3, with an average bag size of 2.13 for LOST, and 2.17 for CSI.

6.1. Results with the basic system

Class-confusion matrices for the top 16 characters in LOST are shown in Figure 8, before and after applying our ambiguous naming system. The most difficult classes are the ones in which another class is strongly correlated in the ambiguous label confusion matrix. This is consistent with the theoretical bounds we obtained in Section 3, which establish a relation between the class-specific error rate and the class-specific degree of ambiguity ϵ.

Quantitative results are shown in Table 1. We measure error according to average 0-1 loss with respect to hand-labeled groundtruth labeled in 8 entire episodes for LOST. Our model does significantly better than all baseline methods. However, we can achieve further improvement by considering additional cues for naming.

[2] Boosted cascade classifiers of Haar features for the eyes, nose and mouth

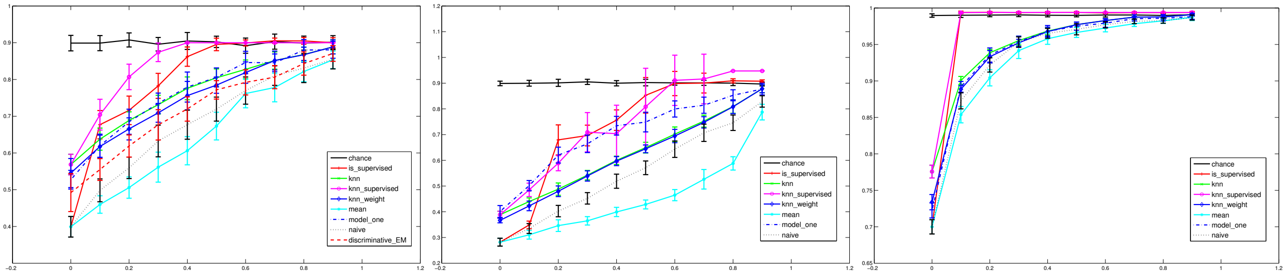

Figure 5. Inductive results on Faces in the Wild, comparing our proposed method (denoted as *mean*) to several baselines. In each case we vary the ambiguity size (x-axis) and report the average error rate (y-axis) and standard deviation over 20 trials. **Left:** balanced dataset using 50 faces for each of the top 10 labels. **Middle:** unbalanced dataset using all faces for each of the top 10 labels. **Right:** unbalanced dataset using up to 100 faces for each of the top 100 labels. See Section 5.2 for details. In all settings, our method outperforms the baselines and previously proposed approaches.

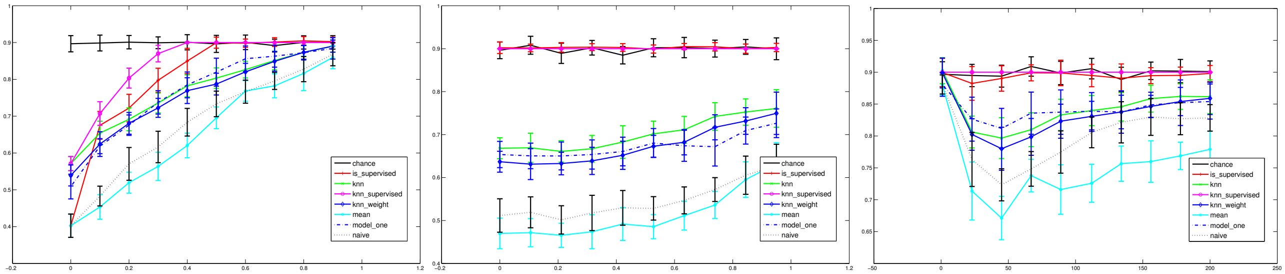

Figure 6. Additional results on Faces in the Wild in different settings. In each case, we report the average error rate (y-axis) and standard deviation over 20 trials as in figure 5. **Left:** increasing ambiguity size, transductive setting (see figure 5 for the corresponding inductive setting). **Middle:** increasing ambiguity degree (Eqn. 1), inductive setting. **Right:** increasing dimensionality, inductive setting.

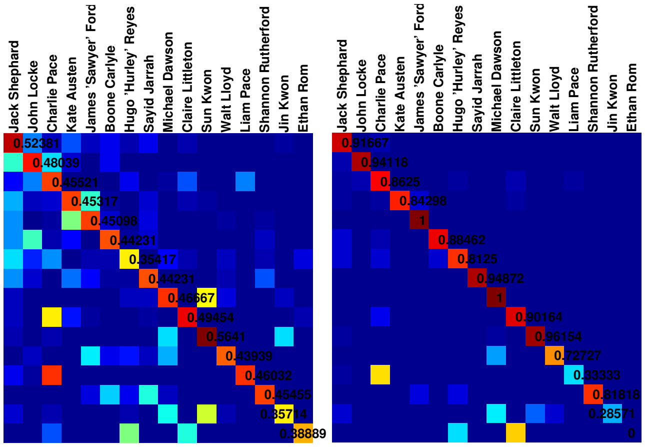

Figure 8. **Left**: Label distribution of top 16 characters in LOST. Element D_{ij} represents the proportion of times class i was seen with class j in the ambiguous bags, and $D\mathbf{1} = \mathbf{1}$. **Right**: Confusion matrix of predictions (without the additional cues) . Element A_{ij} represents the proportion of times class i was classified as class j, and $A\mathbf{1} = \mathbf{1}$. Class priors for the most frequent, the median frequency, and the least frequent characters in LOST are Jack Shephard, 14%; Hugo Reyes, 6%; Liam Pace 1%.

6.2. Additional constraints

We investigate using additional constraints to further improve the performance of our system: mouth motion, grouping constraints and gender. Final misclassification results are reported in Table 1.

LOST (#labels, #eps.)	(8,16)	(16,16)	(32,16)
Naive	14%	16.5%	18.5%
ours ("mean")	10%	14%	17%
ours+constraints	**6%**	**11%**	**13%**

Table 1. Misclassification rates of different methods on TV show LOST. For comparison, other baseline methods' performances for (#labels, #eps.) = $(16, 16)$ are *knn*: 30%; *Model 1*: 44%; *chance*: 53%.

Mouth motion. We use a similar approach to [6] to detect mouth motion during dialog and adapt it to our ambiguous label setting[3]. For a face track x with ambiguous label set Y and a temporally overlapping utterance from a speaker $a \in \{1..L\}$ (after aligning screenplay and closed captions), we restrict Y as follows:

$$Y := \begin{cases} \{a\} & \text{if mouth motion} \\ Y & \text{if refuse to predict or } |Y| = \{a\} \\ Y - \{a\} & \text{if absence of mouth motion} \end{cases} \tag{16}$$

Gender constraints. We introduce a gender classifier to constrain the ambiguous labels based on predicted gender.

[3] Motion or absence of motion are detected with a low and high threshold on normalized cross-correlation around mouth regions in consecutive frames.

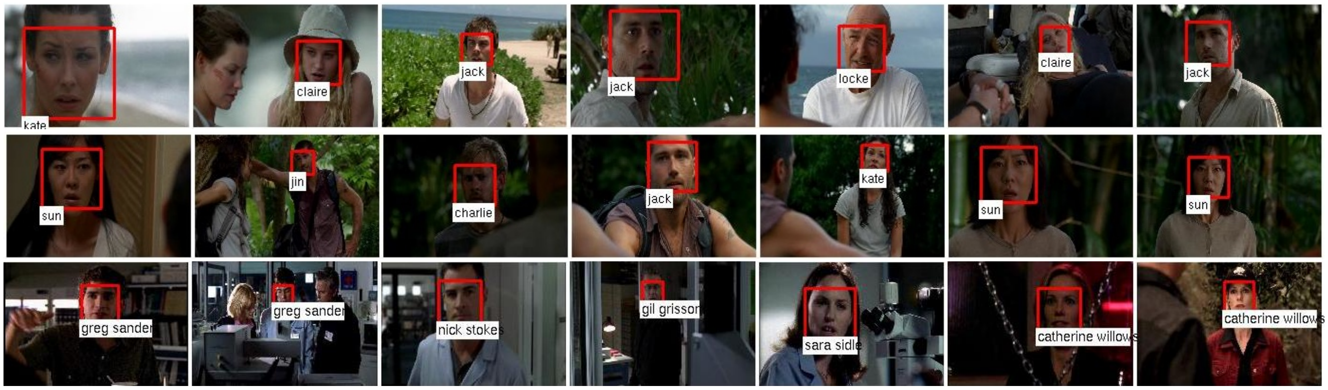

Figure 7. Predictions on LOST and CSI. Incorrect examples are: row 1, column 3 (truth: Boone); row 2, column 2 (truth: Jack).

The gender classifier is trained on a dataset of registered male and female faces, by boosting a set of decision stumps computed on Haar wavelets. Our gender classifier gives a score $\gamma(x)$ for each face track x. We assume known the gender of names mentioned in the screenplay (using automatically extracted cast list from IMDB). We use gender by filtering out the labels that do not match by gender the predicted gender of a face track, if the confidence is greater than a threshold (one threshold for females, one for males, are set on a validation data to achieve 90% precision for each direction of the gender prediction). Thus, we modify ambiguous label set Y as follows:

$$Y := \begin{cases} Y & \text{if gender uncertain} \\ Y - \{a : a \text{ is male}\} & \text{if gender predicts female} \\ Y - \{a : a \text{ is female}\} & \text{if gender predicts male} \end{cases} \quad (17)$$

Grouping constraints. We propose a very simple must-not-link constraint, which states $y_i \neq y_j$ if face tracks x_i, x_j are in two consecutive shots (modeling alternation of shots, common in dialogs). This constraint is active only when a scene has 2 characters. Unlike the previous constraints, this constraint is incorporated as additional terms in our loss function, as in [15].

Ablative analysis. We evaluate with a refusal to predict scheme inspired by [6]. For a given recall rate $r \in [0, 1]$, we extract the $r \cdot m$ most confident predictions and compute error rate on those examples. The confidence is defined as the difference between the best and second best label scores.

Figure 9 is an ablative analysis, showing error rate vs recall curves for different sets of cues. We see that the constraints provided by mouth motion help most, followed by gender and link constraints. The best setting (without using groundtruth) combines the former two cues. Also, we notice, once again, a significant performance improvement of our method over the naive method.

6.3. Qualitative results and Video demonstration

We show examples with predicted labels and corresponding accuracy, for various characters in figures 10, 11, 12, 13 for LOST and figures 14, 15, 16, 17 for CSI. Those results were obtained with the basic system of section 6.1.

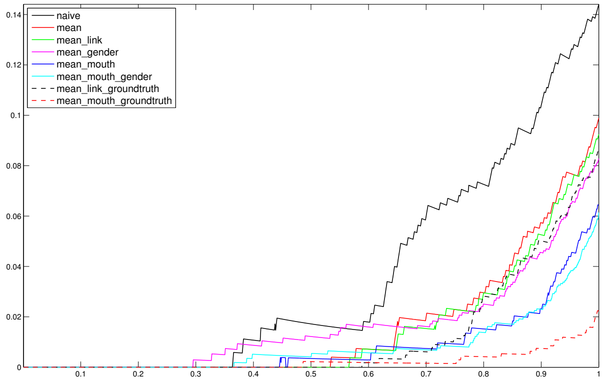

Figure 9. Ablative analysis. x-axis: recall; y-axis: error rate for character naming across 16 episodes of Lost, and the 8 most common labels. We compare our method, *mean*, to the *naive* model and show the effect of adding constraints to our system: gender, mouth motion, and linking constraints (along with their perfect, groundtruth counterparts), described in Section 6.2.

Full-frame detections can be seen in Figure 7. We also propagate the predicted labels of our model to all faces in the same face track throughout an episode. Video results of several episodes can be found at the following website `http://www.youtube.com/user/AmbiguousNaming`.

7. Conclusion

We have presented an effective approach for learning from ambiguously labeled data, where each instance is tagged with more than one label. We show bounds on the classification error, even when all examples are ambiguously labeled. We compared our approach to strong competing algorithms on 2 naming tasks and demonstrated that our algorithm achieves superior performance. We attribute the success of our approach to better modeling of the mutual exclusion between labels, compared to the naive multi-label approach. Moreover, unlike recently published techniques that address similar ambiguously labeled problems, our method does not rely on heuristics and does not suffer

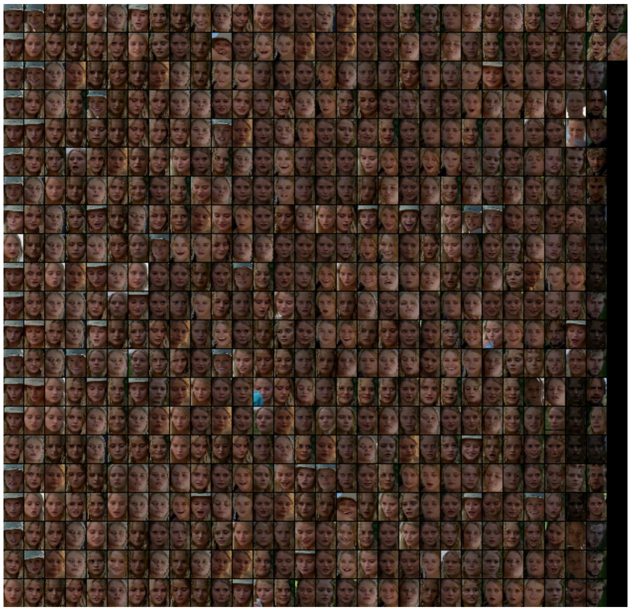

Figure 10. Examples classified as Claire in the LOST data set using our method. Results are sorted by classifier score, in column major format; this explains why most of the errors occur in the last columns. The precision is 97.4%.

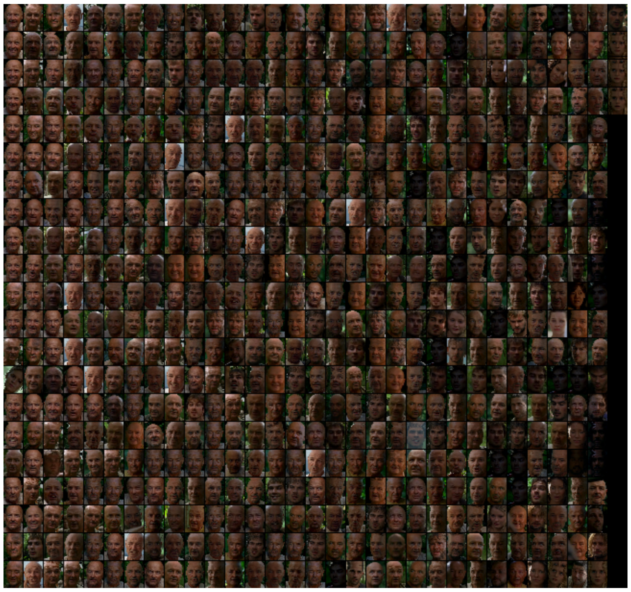

Figure 11. Examples classified as Locke in LOST. The precision is 78.7%.

from local optima of non-convex methods.

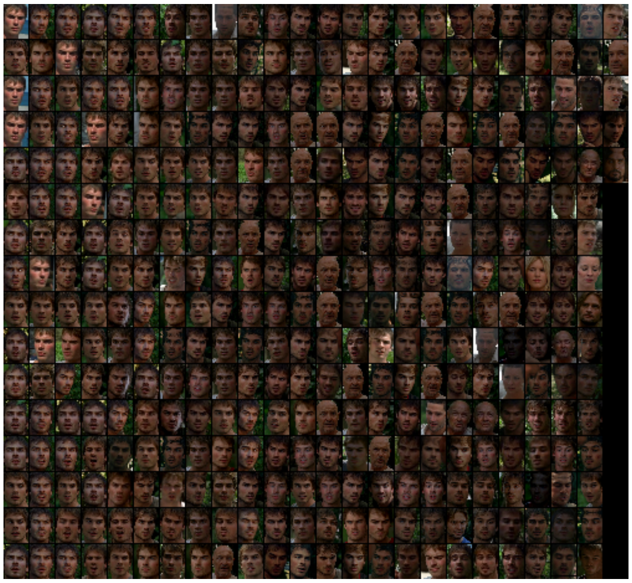

Figure 12. Examples classified as Boone in LOST. The precision is 90.1%.

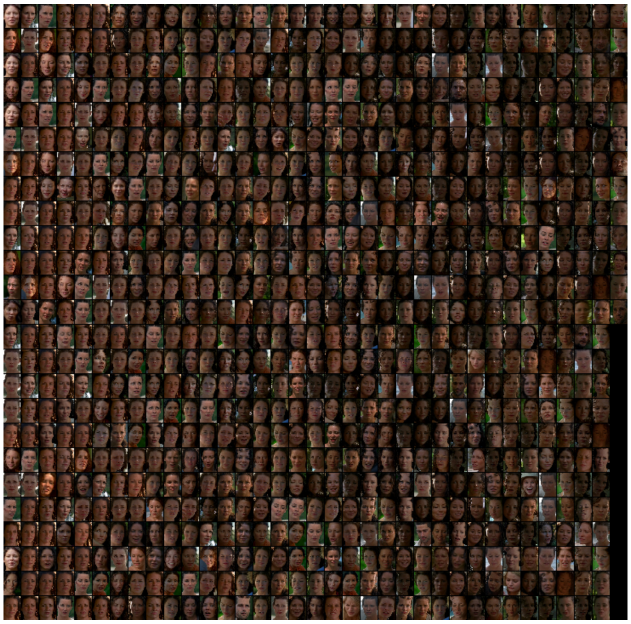

Figure 13. Examples classified as Kate in LOST. The precision is 97.5%.

References

[1] K. Barnard, P. Duygulu, D. Forsyth, N. de Freitas, D. Blei, and M. Jordan. Matching words and pictures. *JMLR*, 2003.

[2] T. Berg, A. Berg, J.Edwards, M.Maire, R.White, Y. Teh, E. Learned-Miller, and D. Forsyth. Names and faces in the news. In *CVPR*, pages 848–854, 2004.

[3] P. E. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 1993.

[4] T. Cour, C. Jordan, E. Miltsakaki, and B. Taskar. Movie/script: Alignment and parsing of video and text transcription. In *Proceedings of 10th European Conference on Computer Vision, Marseille, France*, 2008.

[5] P. Duygulu, K. Barnard, J. de Freitas, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a

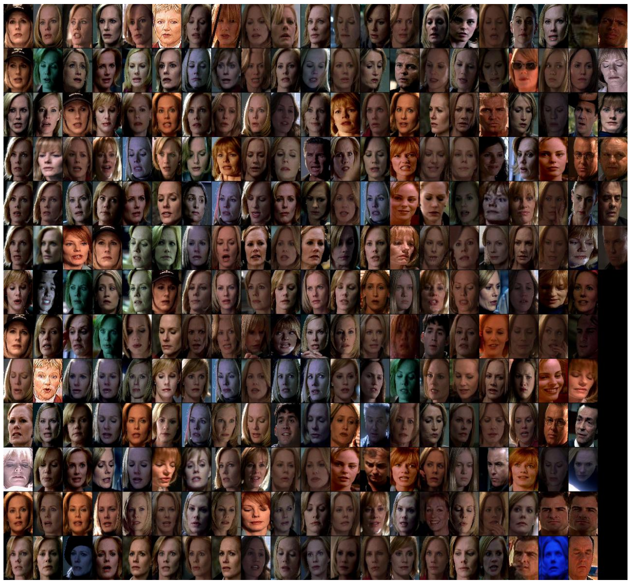

Figure 14. Examples classified as Catherine Willows in CSI. The precision is 85.3%.

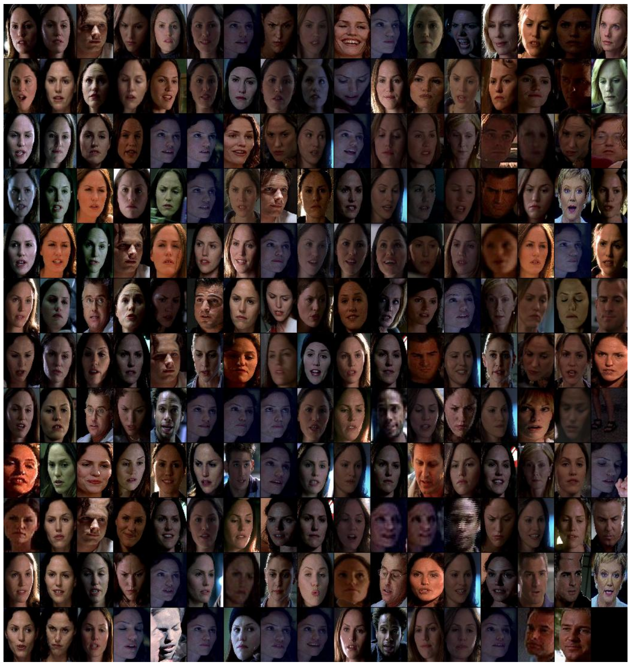

Figure 15. Examples classified as Sara Sidle in CSI. The precision is 78.3%.

fixed image vocabulary. In *ECCV*, pages 97–112, 2002.

[6] M. Everingham, J. Sivic, and A. Zisserman. Hello! My name is... Buffy – automatic naming of characters in tv video. In *BMVC*, 2006.

[7] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *JMLR*, 2008.

[8] A. Gallagher and T. Chen. Using group prior to identify people in consumer images. In *CVPR Workshop on Semantic Learning Applications in Multimedia*, 2007.

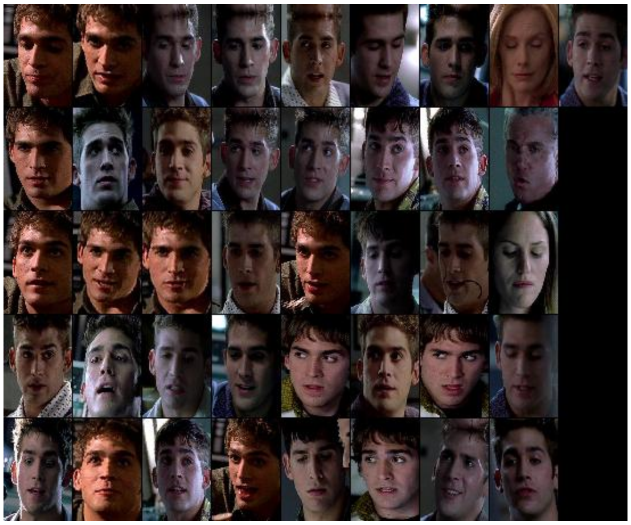

Figure 16. Examples classified as Greg Sanders in CSI. The precision is 92.7%.

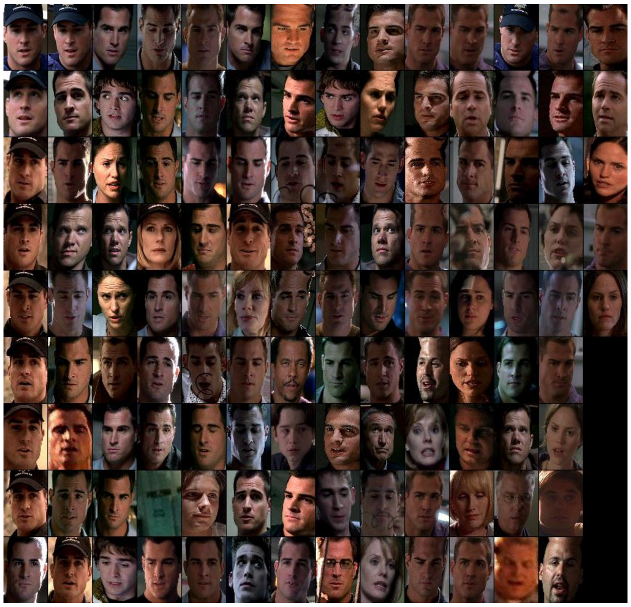

Figure 17. Examples classified as Nick Stokes in CSI. The precision is 66.4%.

[9] G. Huang, V. Jain, and E. Learned-Miller. Unsupervised joint alignment of complex images. In *ICCV*, 2007.

[10] G. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[11] E. Hullermeier and J. Beringer. Learning from ambiguously labeled examples. *Intell. Data Analysis*, 2006.

[12] R. Jin and Z. Ghahramani. Learning with multiple labels. In *NIPS*, pages 897–904, 2002.

[13] D. Ramanan, S. Baker, and S. Kakade. Leveraging archival video for building face datasets. In *ICCV*, 2007.

[14] S. Satoh, Y. Nakamura, and T. Kanade. Name-it: Naming and detecting faces in news videos. *IEEE MultiMedia*, 1999.

[15] R. Yang and A. Hauptmann. A discriminative learning framework with pairwise constraints for video object classification. *PAMI*, 28(4):578–593, 2006.

[16] T. Zhang. Statistical analysis of some multi-category large margin classification methods. *JMLR*, 5, 2004.