Multiple Instance Metric Learning from Automatically Labeled Bags of Faces

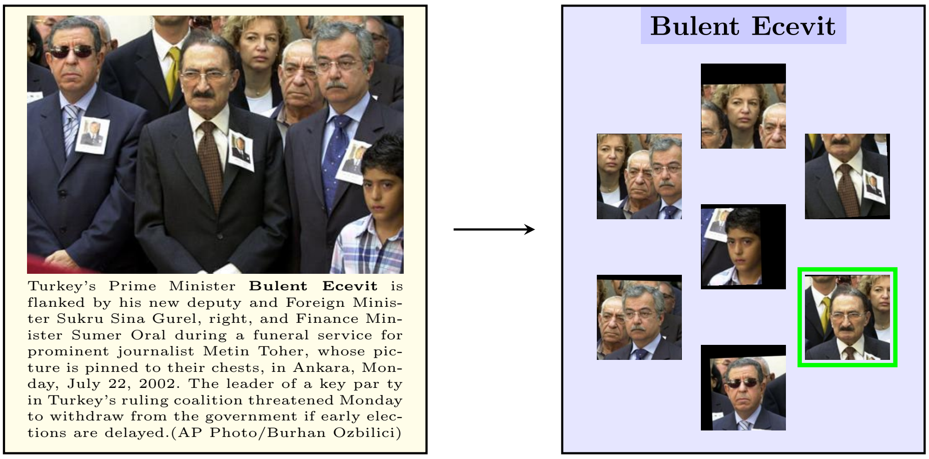
Matthieu Guillaumin Jakob Verbeek Cordelia Schmid

LEAR, INRIA, Grenoble Laboratoire Jean Kuntzmann
firstname.lastname@inria.fr

Abstract. Metric learning aims at finding a distance that approximates a task-specific notion of semantic similarity. Typically, a Mahalanobis distance is learned from pairs of data labeled as being semantically similar or not. In this paper, we learn such metrics in a weakly supervised setting where "bags" of instances are labeled with "bags" of labels. We formulate the problem as a multiple instance learning (MIL) problem over pairs of bags. If two bags share at least one label, we label the pair positive, and negative otherwise. We propose to learn a metric using those labeled pairs of bags, leading to MildML, for multiple instance logistic discriminant metric learning. MildML iterates between updates of the metric and selection of putative positive pairs of examples from positive pairs of bags. To evaluate our approach, we introduce a large and challenging data set, *Labeled Yahoo! News*, which we have manually annotated and contains 31147 detected faces of 5873 different people in 20071 images. We group the faces detected in an image into a bag, and group the names detected in the caption into a corresponding set of labels. When the labels come from manual annotation, we find that MildML using the bag-level annotation performs as well as fully supervised metric learning using instance-level annotation. We also consider performance in the case of automatically extracted labels for the bags, where some of the bag labels do not correspond to any example in the bag. In this case MildML works substantially better than relying on noisy instance-level annotations derived from the bag-level annotation by resolving face-name associations in images with their captions.

1 Introduction

Metric learning is a supervised technique that finds a metric over a feature space that corresponds to a semantic distance defined by an annotator, who provides pairs of examples labeled with their semantic distance (typically zero or one). This semantic distance, in computer vision, might for instance express that two images depict the same object, or that they possess roughly the same layout. Once learned, the metric can be used in many different settings, *e.g.* for k nearest neighbor classification [1] , matching and clustering samples based on the semantic similarity [2, 3], or indexing for information retrieval and data visualization [4, 5].



Turkey's Prime Minister **Bulent Ecevit** is flanked by his new deputy and Foreign Minister Sukru Sina Gurel, right, and Finance Minister Sumer Oral during a funeral service for prominent journalist Metin Toher, whose picture is pinned to their chests, in Ankara, Monday, July 22, 2002. The leader of a key par ty in Turkey's ruling coalition threatened Monday to withdraw from the government if early elections are delayed.(AP Photo/Burhan Ozbilici)

Fig. 1. Viewing news images with captions as a Multiple Instance Learning problem. The label "Bulent Ecevit" is assumed to be valid for at least one face in the face bag. The correct face image for Bulent Ecevit is highlighted in green.

Metric learning has recently received a lot of attention [1, 3, 6–10]. Most methods learn a Mahalanobis metric, which generalizes the Euclidean distance, using a variety of objective functions to optimize the metric. On the one hand, relatively large numbers of labeled pairs of examples are needed to learn Mahalanobis metrics, since the number of parameters scales quadratically with the data dimensionality. On the other hand, increasing the number of labeled pairs will immediately increase the run-time of metric learning algorithms, making large scale applications difficult. Regularization towards the Euclidean metric is often imposed to find a trade-off solution.

Large scale applications regularly arise in the field of computer vision, due to the explosive growth over the last decades of available data resulting from the advent of digital photography, photo sharing websites like Flickr and Facebook, or news media publishing online. Rarely, though, does this data come with clean annotations for the visual content. In an increasing number of cases, additional information relating to the images is nevertheless present, *e.g.* tags in Flickr or Facebook, captions for news images or surrounding text in web pages. Given this observation, a question that naturally arises is whether this massive quantity of weakly annotated data can be used to perform metric learning. Although weak forms of supervision have been considered for a number of computer vision related tasks [11, 12], there is currently more work on metric learning from semi-supervised settings [2, 13] than from noisy and weak supervision [14].

In this paper, we focus on a particular form of weak supervision where data points are clustered in small groups that we call bags. Bags appear naturally in several computer vision settings: for instance, an image can be viewed as a bag of several regions or segments [15] – each of which is described by a feature vector– or a video sequence as a bag of frames [14]. Multiple instance learning (MIL) [16] refers precisely to the class of problems where data instances appear in bags, and each bag contains at least one instance for each label associated with the bag.

The work closest related to our is [17], where the authors learn a metric from MIL data for image auto-annotation. Compared to their settings, though, we will investigate the performance of metric learning when bag labels are noisy, which means that the underlying assumption of MIL will not be true in general: a bag may be assigned a label for which none of the instances is relevant.

More precisely, we focus on the problem of metric learning for face recognition. The goal is to obtain a metric that relates to the identity of a person, despite wide variations in pose, expression, illumination, hair style, etc. In our setting, bags are images and instances are faces appearing in these images, as illustrated in Figure 1. The labels are names automatically extracted from the image caption. As we will see, in this setting the handling of noisy labels can be viewed as a constrained clustering problem. Constrained clustering of faces using noisy name-labels has been considered by various authors [18–22], but these approaches do not integrate this with metric learning, except [2].

The paper is organized as follows: first we describe LDML [3], a recent state-of-the-art metric learning algorithm. Next, we propose a MIL formulation for learning similar types of metrics but on bag-level annotations which are potentially noisy. We refer to it as MildML, for Multiple Instance Logistic Discriminant Metric Learning. Then, we show how LDML can be adapted to infer instance-level annotations from bag-level annotation, and how this can help handle noisy data. In Section 3, we present the *Yahoo! News* data set, and our annotation of it that is publicly available as the *Labeled Yahoo! News* data set, and our feature extraction procedure. Section 4 presents our experimental results, and we conclude in Section 5.

2 Metric Learning From Various Levels of Annotation

In this section we show how Mahalanobis metrics can be learned from strong instance-level annotations to weaker bag-level forms of supervision. A Mahalanobis distance $d_{\mathbf{M}}$ on $\mathbb{R}^D$ generalizes the Euclidean distance, and for $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^D$ it is defined as

$$d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^\top \mathbf{M}(\mathbf{x}_i - \mathbf{x}_j), \tag{1}$$

where $\mathbf{M}$ is a $D \times D$ symmetric positive semidefinite matrix, *i.e.* $\mathbf{M} \in \mathcal{S}_D^+$. Note that $\mathcal{S}_D^+$ is a cone in $\mathbb{R}^{D \times D}$, and therefore is a convex subset of $\mathbb{R}^{D \times D}$.

Below, we first describe LDML, a recent supervised metric learning method in Section 2.1, and modify it to learn low-rank matrices $\mathbf{M}$. In Section 2.2 we introduce MildML, a MIL extension of LDML that can handle noisy bag-level annotations. Then, in Section 2.3 we cast MIL metric learning as an joint metric learning and constrained clustering problem.

2.1 Supervised Metric Learning

In fully supervised settings, data points $\mathbf{x}_i$ are manually associated with their true class labels coded by a binary vector $y_i \in \{0,1\}^C$, where C is the number of classes. Let us denote $\mathbf{X}$ the $D \times N$ matrix whose columns are the data vectors $\mathbf{x}_i$, and $\mathbf{Y} = [y_i] \in \mathbb{R}^{C \times N}$ the label matrix. Typically, exactly one component of y_i equals 1 and all the other equal 0.

This is the classical metric learning setup, and it has been extensively studied [1, 3, 10]. Here, we focus on Logistic Discriminant Metric Learning (LDML) [3], which maximizes the concave log-likelihood $\mathcal{L}$ of a logistic discriminant model. Considering the convexity of $\mathcal{S}_D^+$, the optimization problem is convex, and can be solved for example using projected gradient descent [23]. The objective of LDML is:

$$\underset{\mathbf{M},b}{\text{maximize}} \quad \mathcal{L} = \sum_{i,j} t_{ij} \log p_{ij} + (1 - t_{ij}) \log(1 - p_{ij}), \tag{2}$$

where t_{ij} denotes the equality of labels y_i and y_j, *i.e.* $t_{ij} = y_i^\top y_j$, and

$$p_{ij} = p(y_i = y_j | \mathbf{x}_i, \mathbf{x}_j, \mathbf{M}, b) = \sigma(b - d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j)), \tag{3}$$

where $\sigma(z) = (1 + \exp(-z))^{-1}$ is the sigmoid function, and the bias b acts as a threshold on the distance value to decide the identification of a new data pair.

We now modify LDML to learn metrics $\mathbf{M}$ of fixed low rank, which reduces the number of free parameters and thus avoids over-fitting. As constraints on the rank of $\mathbf{M}$ are non-convex, we can no longer use methods for convex optimization. Instead, we choose to define $\mathbf{M} = \mathbf{L}^\top \mathbf{L}$, where $\mathbf{L}$ is a $d \times D$ matrix, which ensures that $\mathbf{M}$ is a positive semidefinite matrix of rank d. We now optimize $\mathcal{L}$ w.r.t. $\mathbf{L}$ using gradient descend, and resort to multiple random initializations to avoid poor local optima. The gradient of $\mathcal{L}$ with respect to $\mathbf{L}$ equals

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \mathbf{L}} &= \mathbf{L} \sum_{i,j} (t_{ij} - p_{ij})(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^\top & (4)\\
&= 2\mathbf{L} \sum_i \mathbf{x}_i \Big(\Big(\sum_j t_{ij} - p_{ij} \Big) \mathbf{x}_i^\top - \sum_j (t_{ij} - p_{ij}) \mathbf{x}_j^\top \Big) & (5)\\
&= 2\mathbf{L}\mathbf{X}\mathbf{H}\mathbf{X}^\top, & (6)
\end{aligned}$$

where $\mathbf{H} = [h_{ij}] \in \mathbb{R}^{N \times N}$ with $h_{ii} = \sum_{j \neq i}(t_{ij} - p_{ij})$ and $h_{ij} = p_{ij} - t_{ij}$ for $j \neq i$. The gradient as in Equation 6 can be computed in complexity $O(N(N+D)d)$ which, since $d \ll D$, is significantly better than the original LDML projected gradient, whose complexity is $O(N(N+D)D + D^3)$.

Note that the rows of $\mathbf{L}$ can be restricted to be in the span of the columns of $\mathbf{X}^\top$. This is possible since this is true for the gradient (Equation 6) and since the Mahalanobis distance over the training data is invariant to perturbations of $\mathbf{L}$ in directions outside the span of $\mathbf{X}$. Hence, using $\mathbf{L} = \mathbf{A}\mathbf{X}^\top$, we can write the Mahalanobis distance in terms of inner products between data points, which allows us to use kernel functions to perform non-linear LDML like was done in [9]. Straightforward algebra shows that to learn the coefficient matrix $\mathbf{A}$ we simply replace $\mathbf{X}$ with the kernel matrix $\mathbf{K}$ in the learning algorithm, which will then output the optimized matrix $\mathbf{A}$. In Section 4 we only report results using linear LDML; preliminary results using polynomial kernels did not show improvements over linear LDML.

2.2 Multiple Instance Metric Learning

We now consider the case where the individual labels y_i are unknown. Instead of supervision on the level of single examples, or pairs of examples, we assume here that the supervision is provided at the level of pairs of bags of examples. This naturally leads to a multiple instance learning formulation of the metric learning problem, which we refer to as MildML, for Multiple Instance Logistic Discriminant Metric Learning.

Let us denote a bag of examples as $\mathcal{X}_d = \{\mathbf{x}_1^d, \mathbf{x}_2^d, \dots, \mathbf{x}_{N_d}^d\}$, where N_d is the number of examples in the bag. The supervision is given by labels $t_{de} \in \{0, 1\}$ that indicate whether for a pair of bags $\mathcal{X}_d$ and $\mathcal{X}_e$ there is at least one pair of examples $(\mathbf{x}_1, \mathbf{x}_2) \in \mathcal{X}_d \times \mathcal{X}_e$ such that $\mathbf{x}_1$ and $\mathbf{x}_2$ belong to the same class. If there is such a pair of examples then $t_{de} = 1$, and $t_{de} = 0$ otherwise.

The objective in Equation 2 is readily adapted to the MIL setting by extending the definition of the distance to compare bags [17] with:

$$d_{\mathbf{M}}(\mathcal{X}_d, \mathcal{X}_e) = \min_{\mathbf{x}_1 \in \mathcal{X}_d, \mathbf{x}_2 \in \mathcal{X}_e} d_{\mathbf{M}}(\mathbf{x}_1, \mathbf{x}_2). \tag{7}$$

which, using $p_{de} = \sigma(b - d_{\mathbf{M}}(\mathcal{X}_d, \mathcal{X}_e))$, leads to the following optimization:

$$\underset{\mathbf{M}, b}{\text{maximize}} \quad \mathcal{L} = \sum_{d,e} t_{de} \log p_{de} + (1 - t_{de}) \log(1 - p_{de}). \tag{8}$$

This objective makes bags that share a label closer, and pushes bags that do not share any label apart. For a negative pair of bags, all the pairs of instances that can be made from these two bags are pushed apart since the pair of examples with minimum distance is.

We optimize the objective iteratively by alternating (i) the pair selection by the min operator for a fixed metric, and (ii) optimizing the metric for a fixed selection of pairs. The optimization in the second step is exactly of the same form as the optimization of the low-rank version of LDML presented in the previous section. For a given selection of pairs, we perform only one line search in the direction of the negative gradient, such that the pair selection is performed for each computation of the gradient. This way we do not spend many gradient steps optimizing the metric for a selection of pairs that might be inaccurate.

Note that MildML does not try to specifically assign labels to instances, and instead for each pair of bags only a single pair of instances is used to learn the metric. The benefit is that this single pair is robust to noise in the data, but the drawback is that many pairs of examples are lost, especially the negative ones occurring inside a bag, which may impact the quality of the learned metric.

2.3 Estimating Instance Labels from Bag-level Labels

In this section we consider the setting where we have partial knowledge of the labels of the instances in a bag $\mathcal{X}_d$, given by a label vector $y_d \in \{0,1\}^C$, where $y_d^{(n)} = 1$ indicates that the bag contains at least one example of class n. This setting is also known as Multiple Instance Multiple Label Learning (MIML). MildML is directly applicable in this case by defining $t_{de} = 1$ if $y_d^\top y_e \geq 1$, and $t_{de} = 0$ otherwise. On the other hand, LDML must be adapted to explicitly estimate the labels of the instances in each bag from the bag-level annotation. By estimating the instance labels we obtain a larger set of training pairs suitable for LDML, which may improve over the metric learned by MildML despite the possibly noisy annotation.

To learn a metric in this setting, we optimize the objective in Equation 2 jointly over the metric parameterized by $\mathbf{L}$ and over the label matrix $\mathbf{Y}$ subject to the label constraints given by the bag-level labeling:

$$\underset{\mathbf{Y},\mathbf{L},b}{\text{maximize}} \quad \mathcal{L} = \sum_{i,j} (y_i^\top y_j) \log p_{ij} + (1 - y_i^\top y_j) \log(1 - p_{ij}). \tag{9}$$

Unfortunately, the joint optimization is intractable. For fixed $\mathbf{Y}$, it is precisely the optimization problem discussed in Section 2.1. When optimizing $\mathbf{Y}$ for fixed $\mathbf{L}$ and b, we can rewrite the objective function as follows:

$$\mathcal{L} = \sum_{i,j} (y_i^\top y_j)(\log p_{ij} - \log(1 - p_{ij})) + c = \sum_{i,j} w_{ij}(y_i^\top y_j) + c, \tag{10}$$

where $c = \sum_{ij} \log(1 - p_{ij})$ and $w_{ij} = b - d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j)$ are constants. This optimization problem is NP-hard, and we therefore have to resort to approximate optimization techniques.

Observing that the only non-constant terms in Equation 10 are those for data points in the same class, we can rewrite the objective for a particular instantiation of $\mathbf{Y}$ as

$$\underset{\mathbf{Y}}{\text{maximize}} \sum_{n=1}^{C} \sum_{i \in \mathcal{Y}_n} \sum_{j \in \mathcal{Y}_n} w_{ij}, \tag{11}$$

where $\mathcal{Y}_n$ is the set of indices of instances that are assigned to class n, *i.e.* $\mathcal{Y}_n = \{i | y_i^{(n)} = 1\}$. Equation 11 reveals that we are solving a constrained clustering problem: we have to assign the instances to clusters corresponding to the classes so as to maximize the sum of intra-cluster similarities w_{ij}. The non-zero entries in the bag-level labels y_d define the subset of possible clusters for the instances in a bag $\mathcal{X}_d$. If we have $y_d^\top y_e = 0$ for a pair of bags, this implies cannot-link constraints between all instance pairs that can be constructed from these bags.

To obtain an approximate solution for $\mathbf{Y}$ we perform a form of local optimization. The label optimization is initialized by assigning all instances in a bag to each permissible class according to the bag label. We then maximize $\mathcal{L}$ w.r.t. the labels of the instances in each bag in turn, also enforcing that each instance is assigned to exactly one class. The optimization at bag level can be done exactly and efficiently using bipartite graph matching [24].

In our experiments we compare the method presented in the current section and MildML. Since both optimize an objective function based on LDML, differences in performance will be due to the strategy to leverage the bag-level labels: either by selecting a single pair of instances for each pair of bag (MildML) or by inferring instance level labels.

The method presented in this section is similar to the one presented in [17]. That work also infers instance level labels to perform metric learning in a MIL setting. However, it is based on estimating prototypes, or cluster centers, for each of the classes. The objective then tries to ensure that for each bag and each class label of the bag, there is at least one instance of the bag close to one of the centers of the class. A second term in the objective function forces the centers of different classes to be maximally separated. The optimization scheme is relatively complex, as in each iteration it involves minimizing a non-convex cost function. Due to this complexity and the fact that we are mainly interested in comparing the different strategies to leverage the bag-level annotation, we do not include [17] in our experimental evaluations.

3 Dataset and Feature Extraction

The *Yahoo! News* database was first introduced by Berg *et al.* [19], and was gathered in 2002–2003. It consists of news images and their captions describing the event appearing in the image. We produced a complete ground-truth annotation of the *Yahoo! News* database, which extends the annotation provided by the *Labeled Faces in the Wild*[1] [25]. Our annotation not only includes more face detections and names, but also indicates which faces were detected in the same image, and which names were detected in the caption of that image. We coin our data set *Labeled Yahoo! News*.[2]

Face and Name Detection. We applied the Viola-Jones face detector face detector on the complete *Yahoo! News* database to collect a large number of faces. The variations in appearances with respect to pose, expression, and illumination are wide, as shown in Figure 2. We kept all the detections, including the incorrect ones. In order to collect labels for the faces detected in each image, we ran a named entity recognition system [26] over the image captions. We also used the set of names from the *Labeled Faces in the Wild* data set as a dictionary for finding names in captions. Most often, the putative face labels collected in this manner include the correct name for all faces in an image, although this is not always true. Detected names without corresponding face detections are more common in the data set.

Dataset Annotation. Documents with no detected faces or names were removed, and we manually annotated the 28204 remaining documents for the correct association between detected faces and detected names. For faces detections

[1] Available online: `http://vis-www.cs.umass.edu/lfw/index.html`

[2] Available at `http://lear.inrialpes.fr/data/` together with the facial features.

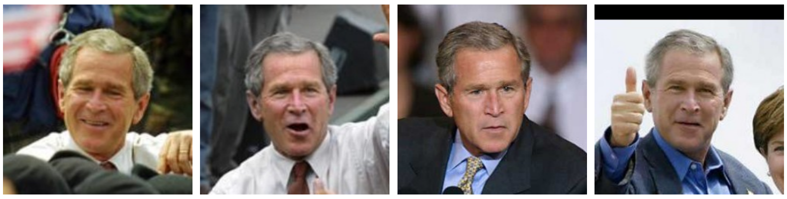


Fig. 2. Examples of wide appearance variations in *Labeled Yahoo! News*.

that are not matched to a name, the annotation indicates whether (a) it is a false detection (not a face), (b) it depicts a person whose name is not in the caption, or (c) it depicts a person whose name was missed by the name detector.

Likewise, for names that are not assigned to a face, the annotation indicates whether the face is present in the image but was missed by the detector. Finally, we also annotate the document when an undetected face matches an undetected name. Illustrations of resulting bags are given in Figure 1 and Figure 3.

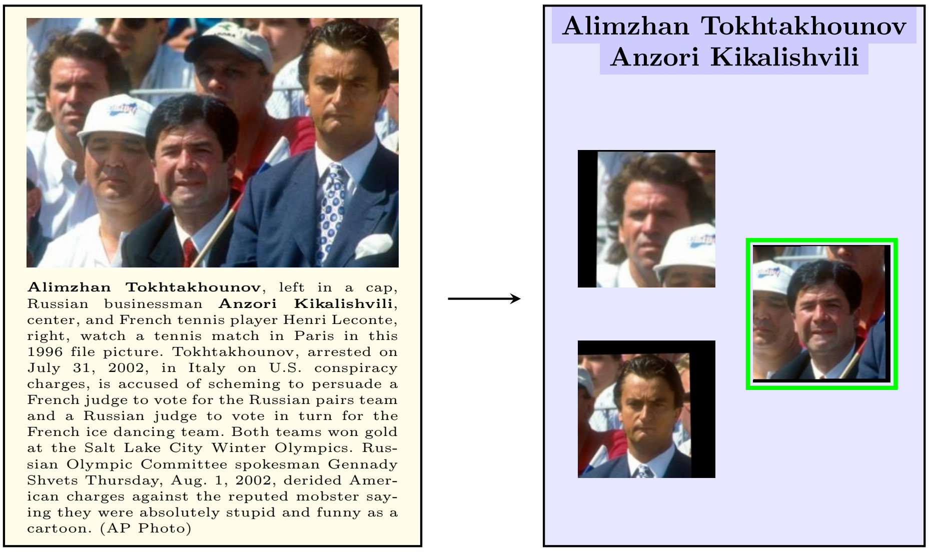
Definition of Test and Train Sets. In order to divide this data set into completely independent training and test sets, we have proceeded the following way. Given the 23 person queries used in [24, 27, 28], the subset of documents containing these names is determined. This set is extended with documents containing "friends" of these 23 people, where friends are defined as people that co-occur in at least one caption [28]. This forms set A. From the remaining set of documents we discard the 8133 ones that contain a name or a face from any person appearing in set A, such that it is now completely disjoint of set A.

Set A contains 9362 documents, 14827 faces and 1072 different names in the captions: because of the specific choice of queries, it has a strong bias towards politicians. Set B contains 10709 documents, 16320 faces and 4801 different people, relatively many athletes. The average number of face images for each person is significantly different between the two sets. Due to these differences between the two sets, we report performance by averaging the results obtained from training on either set and testing on the other.

Facial Feature Extraction. We computed a feature vector for each face detection in the following manner. First, we used the face alignment procedure of [29] to reduce effects due to differences in scale and orientation of the detected faces. Then, nine facial features are detected using the method of [20]. Around each of these nine points we extracted 128-d SIFT descriptors [30] on 3 different scales as in [3]. This results in 3456-d descriptors for every detected face.

4 Experimental results

In this section we present our experimental results, and compare our different methods to learn metrics from weak supervision. The metrics are evaluated for two tasks: verification and clustering.



Alimzhan Tokhtakhounov, left in a cap, Russian businessman **Anzori Kikalishvili**, center, and French tennis player Henri Leconte, right, watch a tennis match in Paris in this 1996 file picture. Tokhtakhounov, arrested on July 31, 2002, in Italy on U.S. conspiracy charges, is accused of scheming to persuade a French judge to vote for the Russian pairs team and a Russian judge to vote in turn for the French ice dancing team. Both teams won gold at the Salt Lake City Winter Olympics. Russian Olympic Committee spokesman Gennady Shvets Thursday, Aug. 1, 2002, derided American charges against the reputed mobster saying they were absolutely stupid and funny as a cartoon. (AP Photo)

Fig. 3. On the left, example document from the *Labeled Yahoo! News* data set with the detected faces and labels on the right. Here, the automatic annotation is incorrect because the face of Alimzhan Tokhtakhounov was not detected. The correct face image for Anzori Kikalishvili is highlighted in green.

4.1 Metrics for Verification

Experimental Protocol. In the face verification task we have to classify a pair of faces as representing the same person or not. Using our *Labeled Yahoo! News* data set, and following the evaluation protocol of [25], we sample 20000 face pairs of both sets A and B, approximately half of which are positives and half are negatives. Note that we can not use the same test set as *Labeled Faces in the Wild* because of overlap between test and train sets in this case. We measure average precision (AP) obtained at ten evenly spaced values of recall, and compute the mean AP (mAP) of (a) training the metric on set A and testing on set B's pairs and (b) training the metric on set B to classify the pairs of set A.

Experimental Results. We study the performance of metric learning for different levels of supervision as described in Section 2. As baseline methods we consider the L2 metric in the original space and after applying PCA to reduce the dimensionality. We compare the following settings for learning metrics:

(a) Instance-level manual annotations. This setting is only applicable to LDML, which should be an upper-bound on performance.
(b) Bag-level manual annotations. This setting is applicable directly to MildML, and indirectly to LDML, using instance-level annotations obtained by applying constrained clustering using the L2 metric to define the face similarities.
(c) Bag-level automatic annotations. Here, the labels are noisy, since the names in the caption do not necessarily correspond to faces detected in the images.

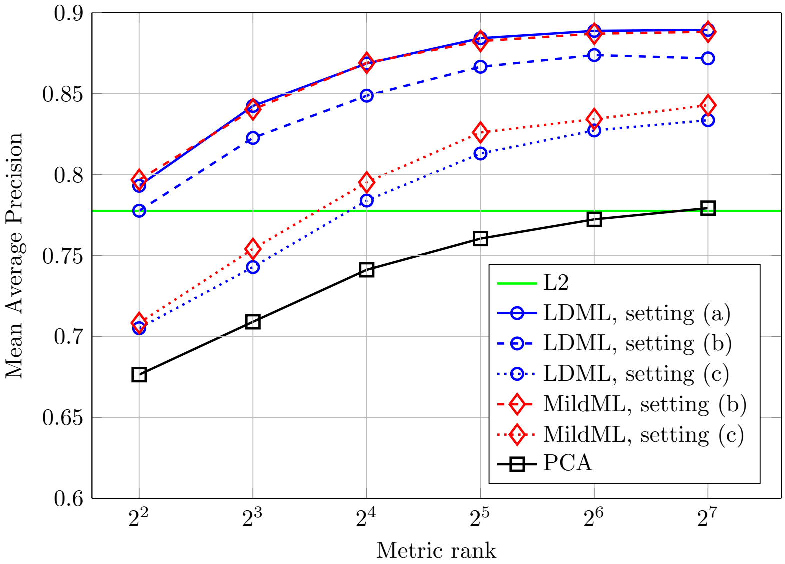


Fig. 4. Mean average precision for L2, PCA, LDML and MildML for the three settings (a), (b) and (c) described in the text, when varying the metric rank.

In Figure 4, we report the performance of PCA, LDML and MildML for a wide range of dimensionalities and for the three settings (a), (b) and (c). As we increase the rank from $d = 4$ to $d = 128$, we can see that the different methods reach a plateau of performance. For LDML with the instance-level annotations (a), the plateau is attained approximately at $d = 32$, with a performance of 88.4% of mAP, which is substantially above L2 and PCA metrics (77.9%).

When learning from manual bag-level annotations (b), we can still learn effective metrics: MildML and LDML are still substantially better than the L2 and PCA metrics. Moreover, MildML matches the performance of the fully supervised LDML on the entire range of metric ranks, with at most 0.6% of improvement for $d = 4$ and 0.2% of decrease at $d = 8$. Notably, MildML outperforms the constrained clustering version of LDML using the same annotation (b), also over the range of metric ranks, by around 2 points.

When using the fully automatic annotation (c), performance drops for both methods, which is understandable since the labels are now noisy. For $d \geq 16$, the performance is still better than L2 and PCA. Also in this setting MildML performs best, reaching 84.3% for 128 dimensions. This score is closer to the fully supervised LDML (89.0%) than to the Euclidean distance (77.8%) or PCA (77.9%) for the same rank. Still, there is a significant gap between the supervised learning and the learning from automatically generated labels, and it appears that this gap narrows from low to higher dimensions: from 8.8% at $d = 4$ to 4.5% at $d = 128$ between the two levels of supervision for MildML.

Setting (b) Rank	4	8	16	32	64	128
LDML	77.8%	82.3%	84.9%	86.7%	87.4%	87.2%
LDML*	76.6%	82.4%	84.8%	86.5%	87.0%	87.0%
Setting (c) Rank	4	8	16	32	64	128
LDML	70.5%	74.3%	78.4%	81.3%	82.7%	83.4%
LDML*	68.1%	73.0%	76.9%	79.2%	80.8%	81.3%

Table 1. Comparison of mean average precision on the *Labeled Yahoo! News* data set for LDML and LDML* metrics. The two tables correspond to annotation settings (b) and (c), respectively. Please refer to the text for more details.

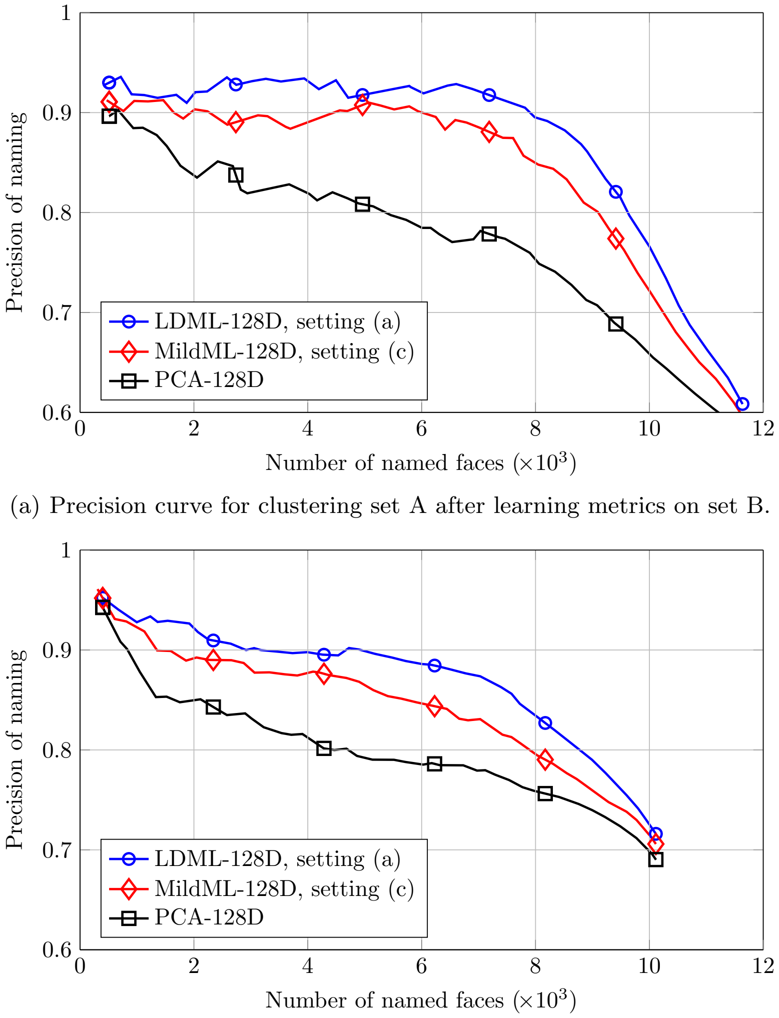
Finally, we also considered a variant of LDML which re-estimates the instance level labels using the current metric, and iterates until convergence. We refer to this variant as LDML*. As shown in Table 1, it has little influence on performance with the manual bag-level annotation of setting (b), at the cost of a much higher training time. On setting (c), the performance drops consistently by around 2%. We conclude that the noisy annotations penalize the clustering significantly. Remarkably, [17] also relies on data clustering while MildML does not.

4.2 Metrics for Constrained Clustering

Experimental Protocol. In our second set of experiments, we assess the quality of the learned metrics for constrained clustering. We use the clustering algorithm described in Section 2.3 on one set of *Labeled Yahoo! News* after learning a metric on the other set. Note, the threshold b in Equation 11 directly influences the number of faces that are indeed associated to a label, *i.e.* named by our algorithm. Therefore, we can measure the precision (*i.e.* the ratio of correctly named faces over total number of named faces) of the clustering procedure for various numbers of named faces by varying the value of b. The curve is approximated on a reasonable range of named faces using a dichotomic search on the threshold value to obtain 50 approximatively evenly spaced points.
Experimental Results. We study the performance of metric learning for different levels of supervision as described in Section 2, while varying the parameter of the clustering algorithm. As a baseline method we consider PCA with 128 dimensions, which performs comparably to the L2 metric. In addition to PCA, we compare the following two learned metrics:

1. The fully supervised 128D LDML (which is comparable in performance to the 128D MildML learned from manual bag-level supervision).
2. The 128D MildML learned from automatically labeled bags of faces.

In Figure 5, we show the naming precision of those three metrics for the two sets: Figure 5(a) for clustering faces of set A, and (b) for set B. First, we notice that the clustering algorithm which associates each instance with a label is efficient, and is able to name several thousand faces with a precision above 80%.



(a) Precision curve for clustering set A after learning metrics on set B.

(b) Precision curve for clustering set B after learning metrics on set A.

Fig. 5. Precision of the clustering algorithm on set A (top) and B (bottom) for three metrics of rank $d = 128$ with the parameter varied, corresponding to a certain percentage of named faces. PCA is an unsupervised metric and performs worst. LDML is fully supervised at instace-level and performs best. MildML is learnt from automatically labeled bags and achieves performance close to the fully supervised metric.

Second, there is a large increase of performance using learned metrics on both sets. LDML performs better than MildML, but the difference (of max. 6.0% between the two curves over the two sets) is smaller than the benefit of using MildML compared to PCA (up to +12.2%).

5 Conclusion

In this paper, we have proposed a Multiple Instance Learning (MIL) formulation of metric learning to allow metric learning from data coming in the form of labeled bags. We refer to it as MildML, for multiple instance logistic discriminant metric learning. We have also shown that it is possible to extend LDML, a instance-level metric learning method, to learn from the same labeled bags using constrained clustering.

On the large and challenging *Labeled Yahoo! News* data set that we have manually annotated, we show that our proposed MildML approach leads to the best results when using bag-level labels. When the bag-level labels are noise-free, the results are comparable to the case where instance level labels are available. When using noisy bag labels, performance drops, but remains significantly better than that of the alternative methods. It appears that performing clustering to obtain instance-level labels and then learning LDML on the labeled examples does not perform well. The (costly) LDML* procedure that iterates metric learning and instance label assignment does not remedy this problem.

In conclusion, we have shown that effective metrics can be learned from automatically generated bag-level labels, underlining the potential of weakly supervised methods. In future work we will consider learning algorithms that scale linearly with the number of data points, allowing learning from much larger data sets. Using larger data sets we expect the difference in performance between weakly supervised and fully supervised learning methods to diminish further.

References

1. Weinberger, K., Blitzer, J., Saul, L.: Distance metric learning for large margin nearest neighbor classification. In: NIPS. (2006)
2. Bilenko, M., Basu, S., Mooney, R.: Integrating constraints and metric learning in semi-supervised clustering. In: ICML, ACM (2004) 11
3. Guillaumin, M., Verbeek, J., Schmid, C.: Is that you? Metric learning approaches for face identification. In: ICCV. (2009)
4. Fu, Y., Li, Z., Huang, T., Katsaggelos, A.: Locally adaptive subspace and similarity metric learning for visual data clustering and retrieval. Computer Vision and Image Understanding **110** (2008) 390–402
5. Jain, P., Kulis, B., Dhillon, I., Grauman, K.: Online metric learning and fast similarity search. In: NIPS. (2008)
6. Xing, E., Ng, A., Jordan, M., Russell, S.: Distance metric learning, with application to clustering with side-information. In: NIPS. (2004)
7. Bar-Hillel, A., Hertz, T., Shental, N., Weinshall, D.: Learning a Mahanalobis metric from equivalence constraints. Journal of Machine Learning Research **6** (2005) 937–965

8. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: CVPR. (2005)
9. Globerson, A., Roweis, S.: Metric learning by collapsing classes. In: NIPS. (2006)
10. Davis, J., Kulis, B., Jain, P., Sra, S., Dhillon, I.: Information-theoretic metric learning. In: ICML. (2007)
11. Wang, J., Markert, K., Everingham, M.: Learning models for object recognition from natural language descriptions. In: BMVC. (2009)
12. Laptev, I., Marszałek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: CVPR. (2008)
13. Wang, F., Chen, S., Zhang, C., Li, T.: Semi-supervised metric learning by maximizing constraint margin. In: Conference on Information and Knowledge Management. (2008)
14. Yang, J., Yan, R., Hauptmann, A.: Multiple instance learning for labeling faces in broadcasting news video. In: ACM Multimedia. (2005)
15. Zhou, Z., Zhang, M.: Multi-instance multi-label learning with application to scene classification. In: NIPS. (2007)
16. Dietterich, T., Lathrop, R., Lozano-Perez, T., Pharmaceutical, A.: Solving the multiple-instance problem with axis-parallel rectangles. Artificial Intelligence **89** (1997) 31–71
17. Jin, R., Wang, S., Zhou, Z.H.: Learning a distance metric from multi-instance multi-label data. In: CVPR. (2009)
18. Satoh, S., Kanade, T.: Name-It: Association of face and name in video. In: CVPR. (1997)
19. Berg, T., Berg, A., Edwards, J., Maire, M., White, R., Teh, Y., Learned-Miller, E., Forsyth, D.: Names and faces in the news. In: CVPR. (2004)
20. Everingham, M., Sivic, J., Zisserman, A.: 'Hello! My name is... Buffy' - Automatic naming of characters in TV video. In: BMVC. (2006)
21. Holub, A., Moreels, P., Perona, P.: Unsupervised clustering for Google searches of celebrity images. In: IEEE Conference on Face and Gesture Recognition. (2008)
22. Pham, P., Moens, M.F., Tuytelaars, T.: Linking names and faces: Seeing the problem in different ways. In: Proceedings of ECCV Workshop on Faces in Real-Life Images. (2008)
23. Bertsekas, D.: On the Goldstein-Levitin-Polyak gradient projection method. IEEE Transactions on Automatic Control **21** (1976) 174–184
24. Guillaumin, M., Mensink, T., Verbeek, J., Schmid, C.: Automatic face naming with caption-based supervision. In: CVPR. (2008)
25. Huang, G., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst (2007)
26. Deschacht, K., Moens, M.: Efficient hierarchical entity classification using conditional random fields. In: Proceedings of Workshop on Ontology Learning and Population. (2006)
27. Ozkan, D., Duygulu, P.: A graph based approach for naming faces in news photos. In: CVPR. (2006) 1477–1482
28. Mensink, T., Verbeek, J.: Improving people search using query expansions: How friends help to find people. In: ECCV. (2008)
29. Huang, G., Jain, V., Learned-Miller, E.: Unsupervised joint alignment of complex images. In: ICCV. (2007)
30. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV **60** (2004) 91–110