Large-scale Image Annotation by Efficient and Robust Kernel Metric Learning

Supplementary Material

Zheyun Feng Rong Jin Anil Jain
Department of Computer Science and Engineering, Michigan State University, East Lansing, MI, 48824, USA
{fengzhey, rongjin, jain}@cse.msu.edu

In this supplementary material, we present

1. Proof of Theorem 1.
2. Additional experimental results on average recall and average F1 score over the three benchmark datasets.
3. A comparison of our proposed RKML algorithm and its counterparts RLML and RKMLH over three benchmark datasets. RLML is the linear counterpart of RKML, while RKMLH is the counterpart of RKML which uses binary constraints.
4. Analysis of sensitivity to parameters in RKML, including rank r, m', the number of retained eigenvectors when estimating the semantic similarity, and n_s, the number of sampled images used for Nyström approximation.
5. A comparison of different design choices of the semantic similarity measure between annotations.
6. A comparison of annotation results generated by LMNN [8] with different methods of generating binary constraints.

1. Proof of Theorem 1

We present here the proofs of Theorem 1 in the main paper stated as follows. Denoting the prediction function for the k-th class by $g_k(\cdot)$, *i.e.*, $y_{i,k} = g_k(\mathbf{x}_i)$, and we make the following assumption for $g_k(\cdot)$ in our analysis:

$$\mathbf{A1}: \quad g_k(\cdot) \in \mathcal{H}_\kappa, \quad k = 1, \ldots, m.$$

Assumption **A1** holds if $g_k(\cdot)$ is a smooth function and $\kappa(\cdot, \cdot)$ is a universal kernel [4].

Theorem 1 *Assume* **A1** *holds, and* $\kappa(\mathbf{x}, \mathbf{x}) \leq 1$ *for any* $\mathbf{x}$. *Let* $r < n$ *be a fixed rank, and* $\lambda_1, \ldots, \lambda_n$ *be the eigenvalues of kernel matrix* K/n *ranked in the descending order. For a fixed failure probability* $\delta \in (0, 1)$, *we assume* n *is large enough such that*

$$\lambda_r \geq \lambda_{r+1} + \frac{8}{\sqrt{n}} \ln(1/\delta).$$

Then, with a probability $1 - \delta$, *we have*

$$\|\widehat{T} - T_*(r)\|_2 \leq \varepsilon,$$

where $\|\cdot\|_2$ *is the spectral norm of a linear operator and* ε *is given by*

$$\varepsilon = \frac{8 \ln(1/\delta)/\sqrt{n}}{\lambda_r - \lambda_{r+1} - 8 \ln(1/\delta)/\sqrt{n}}.$$

1.1. Sketched Proof

We first give the sketch of the proof and refer the readers to Section 1.2 for more detailed analysis. We first rewrite T into the following form using the expression of A in (3)

$$\widehat{T}[f](\cdot) = \sum_{k=1}^{m} \widehat{h}_k(\cdot) \langle \widehat{h}_k(\cdot), f(\cdot) \rangle_{\mathcal{H}_\kappa},$$

where $\widehat{h}_k(\cdot) = \sum_{i=1}^{n} \kappa(\mathbf{x}_i, \cdot)[K_r^{-1}\mathbf{y}^k]_i$, and $\mathbf{y}^k \in \mathbb{R}^n$ is the k-th column vector of matrix Y.

Using the definition of $g_k(\cdot)$ and assumption **A1**, as well as the reproducing property of kernel function [5], we have $y_{i,k} = g_k(\mathbf{x}_i) = \langle g_k(\cdot), \kappa(\mathbf{x}_i, \cdot) \rangle_{\mathcal{H}_\kappa}$. Based on these preparations, we develop the following theorem for $\widehat{h}_k(\cdot)$

Theorem 2 *Under assumption* **A1**, *we have*

$$\widehat{h}_k(\cdot) = \sum_{i=1}^{r} \widehat{\varphi}_i(\cdot) \langle \widehat{\varphi}_i(\cdot), g_k(\cdot) \rangle_{\mathcal{H}_\kappa},$$

where $\widehat{\varphi}_i(\cdot), i = 1, \ldots, r$ *are the first* r *eigenfunctions of the linear operator*

$$L_n[f] = \frac{1}{n} \sum_{i=1}^{n} \kappa(\mathbf{x}_i, \cdot) f(\mathbf{x}_i).$$

Using similar analysis as Theorem 2, we can express T_* as

$$T_*[f] = \sum_{k=1}^{m} h_k(\cdot)\langle h_k(\cdot), f(\cdot)\rangle,$$

where $h_k(\cdot) = \sum_{i=1}^{r} \varphi_i(\cdot)\langle \varphi_i(\cdot), g_k(\cdot)\rangle_{\mathcal{H}_\kappa}$, the projection of prediction function $g_k(\cdot)$ into the subspace spanned by $\{\varphi_i\}_{i=1}^r$. Here $\varphi_i(\cdot)$, $i = 1, \ldots, r$ are the first r eigenfunctions of the integral operator

$$L[f] = \mathrm{E}_{\mathbf{x}}\left[\kappa(\mathbf{x}, \cdot) f(\mathbf{x})\right].$$

Therefore the following theorems bound $\|\widehat{T} - T_*\|_2$ and $\|L - L_n\|_2$ by the following two theorems, respectively.

Theorem 3 *Let λ_r and λ_{r+1} be the r-th and $r+1$-th eigenvalues of kernel matrix K. For a fixed failure probability $\delta \in (0, 1)$, assume*

$$\frac{\lambda_r - \lambda_{r+1}}{n} > \|L - L_n\|_2,$$

where $\|\cdot\|_2$ measures the spectral norm of a linear operator. Then, with a probability $1 - \delta$, we have

$$\max_{f \in \mathcal{H}_\kappa} \|(\widehat{T} - T_*)[f]\|_{\mathcal{H}_\kappa} \leq \gamma \|T_*[f]\|_{\mathcal{H}_\kappa},$$

where γ is given by

$$\gamma = \frac{2\|L - L_n\|_2}{(\lambda_r - \lambda_{r+1})/n - \|L - L_n\|_2}.$$

Theorem 4 *[6] Assume $\kappa(\mathbf{x}, \mathbf{x}) \leq 1$. With a probability $1 - \delta$, we have*

$$\|L - L_n\|_{HS} \leq \frac{4\ln(1/\delta)}{\sqrt{n}}.$$

Theorem 1 follows immediately from Theorem 4 and 3.

1.2. Proofs of the Support Theorems

Proof of Theorem 2 : Let $(\lambda_i, \mathbf{u}_i), i = 1, \ldots, n$ be the eigenvalues and eigenvectors of K. Define $U = (\mathbf{u}_1, \ldots, \mathbf{u}_n)$. According to [6], the eigenfunctions of L_n is given by

$$\widehat{\varphi}_i(\cdot) = \frac{1}{\sqrt{\lambda_i}} \sum_{j=1}^{n} U_{j,i} \kappa(\mathbf{x}_j, \cdot).$$

We therefore have

$$\begin{aligned}
&\sum_{i=1}^{r} \widehat{\varphi}_i(\cdot)\langle \widehat{\varphi}_i(\cdot), g_k(\cdot)\rangle_{\mathcal{H}_\kappa} \\
= &\sum_{i=1}^{r} \sum_{a,b=1}^{n} \frac{1}{\lambda_i} \kappa(\mathbf{x}_a, \cdot)\langle \kappa(\mathbf{x}_b, \cdot), g_k(\cdot)\rangle_{\mathcal{H}_\kappa} U_{a,i} U_{b,i} \\
= &\sum_{i=1}^{r} \sum_{a=1} \kappa(\mathbf{x}_a, \cdot) \frac{1}{\lambda_i} U_{a,i} U_{b,i} Y_{b,k} \\
= &\sum_{i=1}^{r} \sum_{a=1} \kappa(\mathbf{x}_a, \cdot) \frac{1}{\lambda_i} U_{a,i} U_{*,i}^\top \mathbf{y}^k \\
= &\sum_{a=1}^{n} \kappa(\mathbf{x}_a, \cdot)[U_r \Sigma_r^{-1} U_r \mathbf{y}^k]_i \\
= &\sum_{a=1}^{n} \kappa(\mathbf{x}_a, \cdot)[K_r^{-1} \mathbf{y}^k]_i.
\end{aligned}$$

Proof of Theorem 3 : Define a linear operator G as

$$G[f] = \sum_{k=1}^{m} g_k(\cdot)\langle g_k, f\rangle_{\mathcal{H}_\kappa}.$$

Define two projection operator $\widehat{P}$ and P as

$$\widehat{P}[f] = \sum_{i=1}^{r} \widehat{\varphi}_i(\cdot)\langle \widehat{\varphi}_i(\cdot), f(\cdot)\rangle_{\mathcal{H}_\kappa},$$
$$P[f] = \sum_{i=1}^{r} \varphi_i(\cdot)\langle \varphi_i(\cdot), f(\cdot)\rangle_{\mathcal{H}_\kappa}.$$

Using G, $\widehat{P}$ and P, we write $\widehat{T}$ and T_* as

$$\widehat{T} = \widehat{P} G \widehat{P}, \quad T_* = PGP.$$

Using the $\sin\Theta$ theorem [7], we have

$$|\widehat{P} - P| \leq \frac{|L - L_n|_2}{\lambda_r(L_n) - \lambda_{r+1}(L)}.$$

Since $\lambda_r(L_n) = \lambda_r/n$, and $\lambda_{r+1}(L) \leq \lambda_{r+1}(L_n) + |L - L_n|_2$, we have

$$|\widehat{P} - P| \leq \frac{|L - L_n|_2}{(\lambda_r - \lambda_{r+1})/n - |L - L_n|_2}.$$

We complete the proof by using the fact

$$|(\widehat{T} - T)[f]|_{\mathcal{H}_\kappa} \leq |(\widehat{P} - P) G \widehat{P}[f]|_{\mathcal{H}_\kappa} + |PG(\widehat{P} - P)[f]|_{\mathcal{H}_\kappa}.$$

2. Additional Experimental Results

In this section ,we report the comparison results between our proposed RKML algorithm and the state-of-the-art approaches for both distance metric learning and image annotation in terms of the average recall and F1 score for the top t annotated tags.

2.1. Comparison of Average Recall on Three Benchmark Datasets

In Figure 1, 2 and 3, we report the average recall for the top t annotated tags obtained by the state-of-the-art non-linear distance metric learning algorithms, linear distance metric learning algorithms and image annotation methods, respectively.

2.2. Comparison of Average F1 Score on Three Benchmark Datasets

In Figure 4, 5 and 6, we report the average F1 score for the top t annotated tags obtained by the state-of-the-art non-linear distance metric learning algorithms, linear distance metric learning algorithms and image annotation methods, respectively.

3. Comparison of RKML with its Linear or Binary Constrained Counterparts

In this section, in order to verify the advantage of using kernel in distance metric learning, we include the the comparison between our proposed RKML algorithm and its linear counterpart RLML. And to illustrate the benefits of the real-valued similarity measure used in our proposed RKML algorithm, we also include the comparison with RKMLH, which adopts the binary constraints used in the baseline distance metric learning algorithms instead of the real-valued ones used in RKML.

Table 1, 2 and 3 present the comparison of RKML, RLML and RKMLH in terms of the average precision for the top t annotated tags.

4. Sensitivity to Parameters in RKML

In this section, we analyze the sensitivity to parameters in RKML, including rank r, m', the number of retained eigenvectors when estimating the semantic similarity, and n_s, the number of sampled images used for Nyström approximation. The experimental results shown in Figure 7 lead to following conclusions. First, while the average accuracy of test images initially improves significantly with increasing rank r, it becomes saturated after certain rank. However, the prediction accuracy of training data increases almost linearly with respect to the rank. Secondly, our RKML algorithm is insensitive to the values of m' and n_s over a wide range.

5. Comparison of Different Design Choices of the Semantic Similarity Measure

We examine the choice of semantic similarity by evaluating the prediction accuracy with varied definition of $\tilde{y}_{i,j}$ in Equation (5). $\tilde{y}_{i,j}$ is actually the product of a local tag weight $l_{i,j}$ that describes the relative occurrence of tag j in image i , and a global weight g_j that describes the relative occurrence of tag j within the entire tag collection. The examined weighting functions [2] are defined as follows in Table 4 and 5.

Binary	$l_{i,j} = 1$ if tag i exists in image j, or else 0.
Term Frequency (TF)	$l_{i,j} = tf_{i,j}$, the occurrences counts of tag j in image i.
Log	$l_{i,j} = \log(tf_{i,j} + 1)$

Table 4. Local weighting functions.

Binary	$g_j = 1$
Normal	$g_j = 1/\sqrt{\sum_i^n tf_{i,j}^2}$
Idf	$g_j = \log_2 \frac{n}{1+df_j}$
Entropy	$g_j = 1 + \sum_i^n \frac{p_{i,j} \log p_{i,j}}{\log n}$, where $p_{i,j} = \frac{tf_{i,j}}{\sum_i^n tf_{i,j}}$

Table 5. Global weighting functions.

AP@t(%)	t=1	t=4	t=7	t=10
Binary-Binary	56 ± 1.01	41 ± 0.57	33 ± 0.49	28 ± 0.45
Binary-Normal	53 ± 1.28	39 ± 0.62	32 ± 0.54	28 ± 0.44
Cosine	56 ± 1.19	41 ± 0.61	33 ± 0.52	28 ± 0.47
TF-IDF	55 ± 1.12	41 ± 0.57	33 ± 0.50	28 ± 0.44
Log-IDF	55 ± 1.12	41 ± 0.57	33 ± 0.50	28 ± 0.44
Log-Entropy	55 ± 1.10	41 ± 0.57	33 ± 0.49	28 ± 0.45

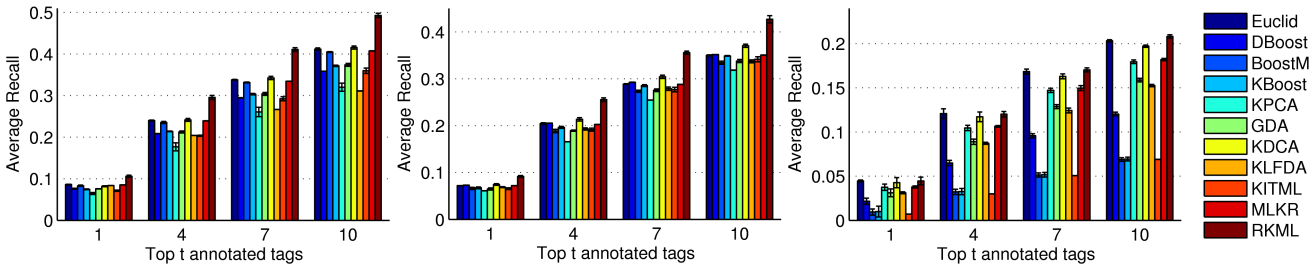
Table 6. Comparison of extensions of RKML with different design choices of semantic similarity for the top t annotated tags on the IAPR TC12 dataset. The leftmost column lists the different weighting methods, where the name before "-" denotes the local weights shown in Table 4 and the name behind "-" indicates the global weights shown in Table 5. "Consine" represents the cosine similarity between tag vectors of two images.

Table 6 shows that different semantic similarity measures, either TF-IDF based weighting or the popular cosine similarity, provide essentially similar performances. We hence adopt the Log-Entropy weighting scheme in our experiments.

6. Comparison of Different Methods of Generating Binary Constraints

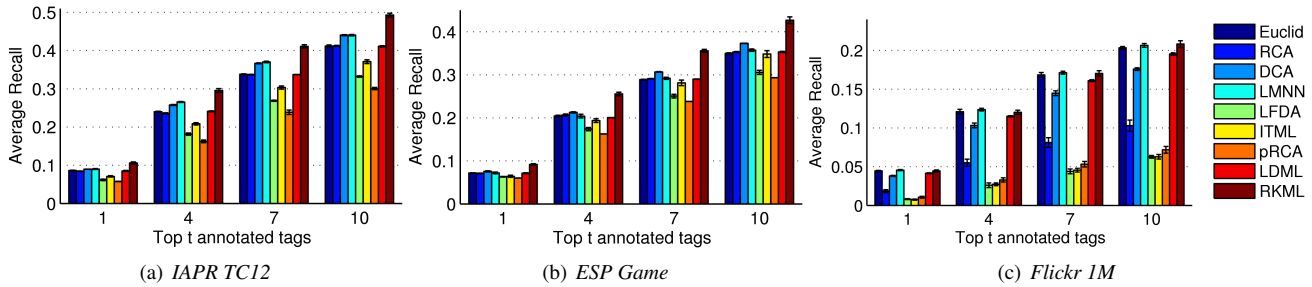
Most DML algorithms were designed for binary constraints. We tried to improve the performance of standard DML algorithms by experimenting with different methods for generating binary constraints. They are listed as follows: (1) Clustering the space of keywords, (2) Generating binary constraints from classification labels[1], (3) Clustering

[1]Flickr1M dataset also includes class assignment labels which is usually used for classification. ESP Game and IAPR TC12 do not have classification labels.



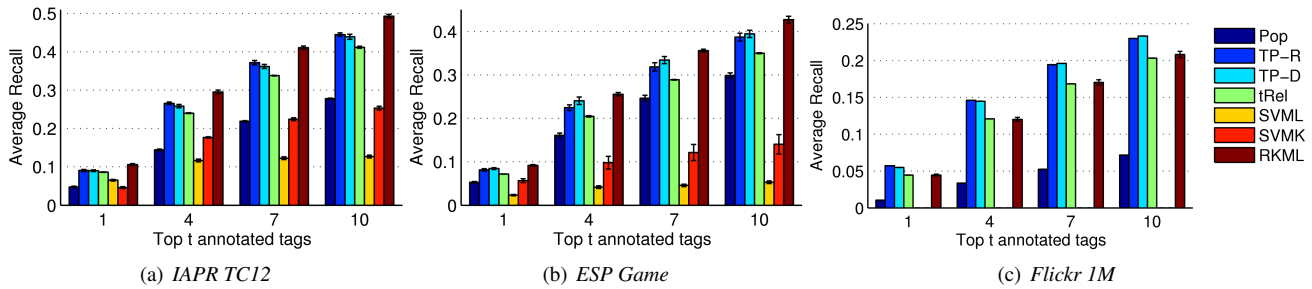
(a) *IAPR TC12* (b) *ESP Game* (c) *Flickr 1M*

Figure 1. Average recall for the top t annotated tags using nonlinear distance metrics.



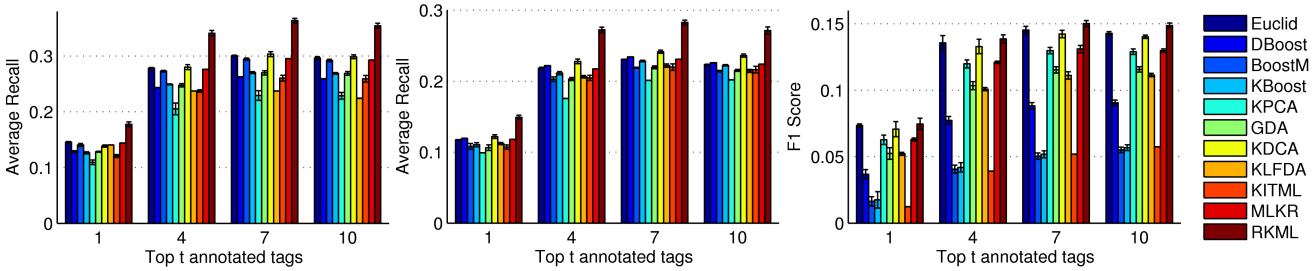
(a) *IAPR TC12* (b) *ESP Game* (c) *Flickr 1M*

Figure 2. Average recall for the top t annotated tags using linear distance metrics.



(a) *IAPR TC12* (b) *ESP Game* (c) *Flickr 1M*

Figure 3. Average recall for the top t annotated tags using different annotation models. SVML and SVMK methods are not included in (c) due to their high computational cost.

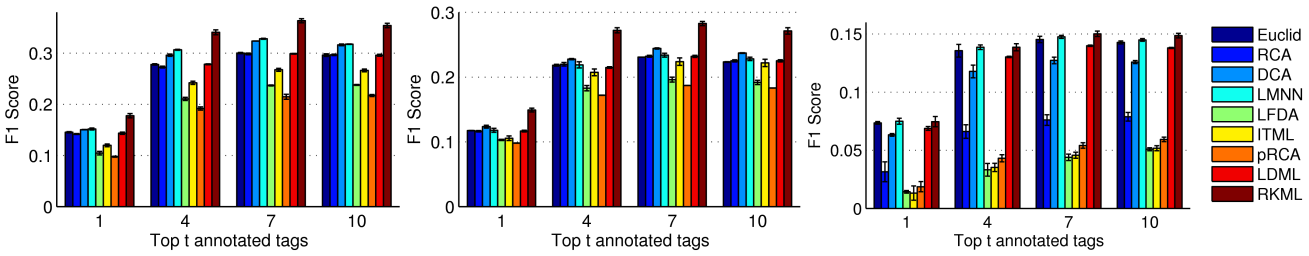


(a) *IAPR TC12* (b) *ESP Game* (c) *Flickr 1M*

Figure 4. Average F1 score for the top t annotated tags using nonlinear distance metrics.

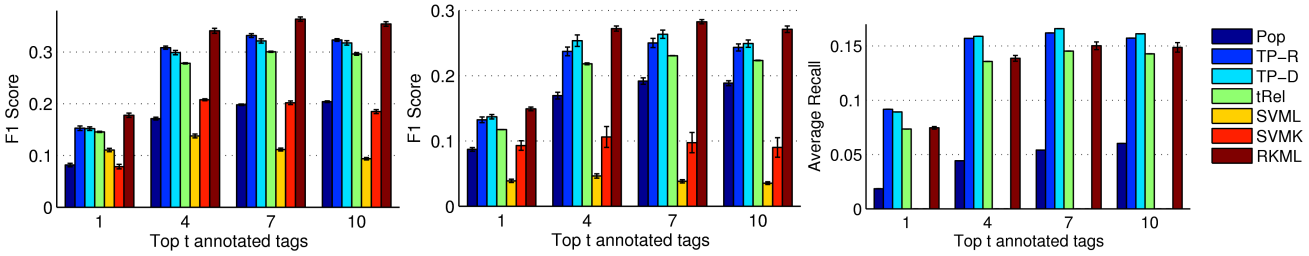
AP@t(%)	t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8	t=9	t=10
RKML	55 ± 1.2	48 ± 0.9	44 ± 0.6	41 ± 0.8	37 ± 0.6	35 ± 0.5	33 ± 0.6	31 ± 0.5	29 ± 0.4	28 ± 0.4
RKMLH	50 ± 1.1	44 ± 0.9	39 ± 0.9	36 ± 0.7	33 ± 0.7	31 ± 0.7	29 ± 0.7	27 ± 0.6	26 ± 0.5	24 ± 0.5
RLML	52 ± 1.3	46 ± 1.2	42 ± 1.0	38 ± 0.8	35 ± 0.7	33 ± 0.6	31 ± 0.5	29 ± 0.5	28 ± 0.4	26 ± 0.4

Table 1. Comparison of various extensions of RKML for the top t annotated tags on the IAPR TC12 dataset.



(a) *IAPR TC12* (b) *ESP Game* (c) *Flickr 1M*

Figure 5. Average F1 score for the top t annotated tags using linear distance metrics.



(a) *IAPR TC12* (b) *ESP Game* (c) *Flickr 1M*

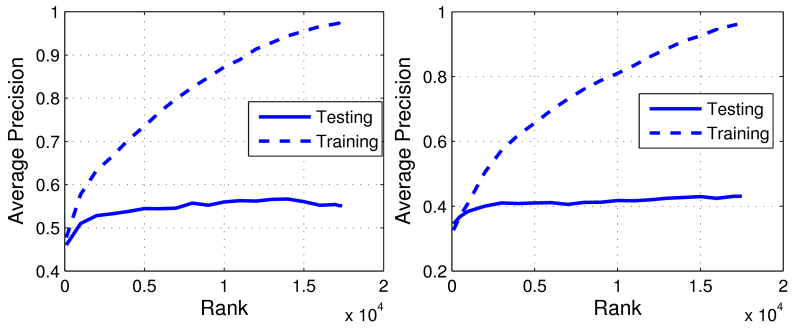
Figure 6. Average F1 score for the top t annotated tags using different annotation models. SVML and SVMK methods are not included in (c) due to their high computational cost.

AP@t(%)	t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8	t=9	t=10
RKML	40 ± 1.1	35 ± 0.5	32 ± 0.4	29 ± 0.5	27 ± 0.4	25 ± 0.4	23 ± 0.3	22 ± 0.4	21 ± 0.4	20 ± 0.4
RKMLH	34 ± 1.0	30 ± 0.5	28 ± 0.5	26 ± 0.4	24 ± 0.4	22 ± 0.3	21 ± 0.3	20 ± 0.3	19 ± 0.3	18 ± 0.3
RLML	36 ± 0.8	31 ± 0.7	28 ± 0.7	26 ± 0.7	24 ± 0.5	22 ± 0.4	21 ± 0.4	20 ± 0.4	19 ± 0.4	18 ± 0.4

Table 2. Comparison of various extensions of RKML for the top t annotated tags on the ESP Game dataset.

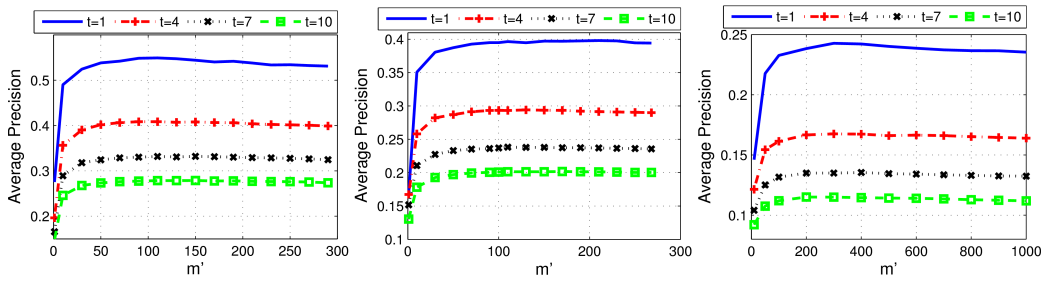
AP@t(%)	t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8	t=9	t=10
RKML	24 ± 0.1	21 ± 0.2	18 ± 0.1	17 ± 0.2	15 ± 0.2	14 ± 0.1	14 ± 0.1	13 ± 0.2	12 ± 0.2	12 ± 0.1
RKMLH	20 ± 0.2	18 ± 0.1	16 ± 0.2	15 ± 0.2	14 ± 0.2	13 ± 0.1	12 ± 0.1	11 ± 0.1	11 ± 0.1	10 ± 0.1
RLML	13 ± 0.3	12 ± 0.2	11 ± 0.2	11 ± 0.1	10 ± 0.06	10 ± 0.05	9.0 ± 0.06	9.0 ± 0.05	8.0 ± 0.06	8.0 ± 0.08

Table 3. Comparison of various extensions of RKML for the top t annotated tags on the Flickr 1M dataset.



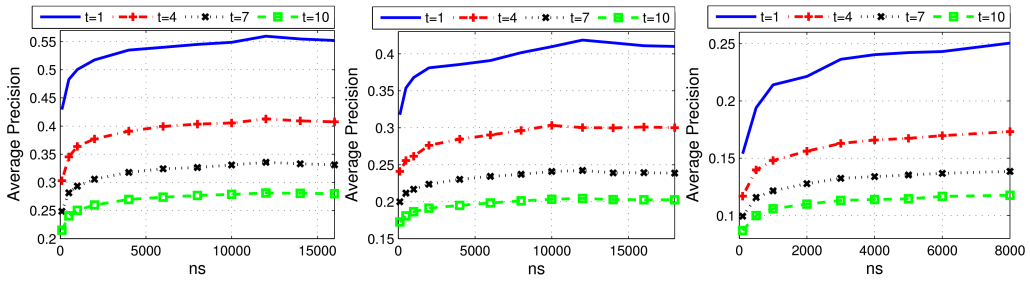
(a) *IAPR TC12* (b) *ESP Game*

Figure 7. Average Precision for the first tag predicted by RKML using different values of rank r. To make the overfitting effect clearer, we turn off the Nyström approximation for IAPR TC12 and ESP Game datasets. Flickr 1M dataset is not included due to its large size ($n = 999,764$). The overfitting only occurs when r approximates to the total number of images, but it is infeasible to apply such a large r in Flickr 1M dataset.



(a) *IAPR TC12* (b) *ESP Game* (c) *Flickr 1M*

Figure 8. Average Precision for the top t tags predicted by RKML using different values of m', the number of retained eigenvectors when estimating the semantic similarity.



(a) *IAPR TC12* (b) *ESP Game* (c) *Flickr 1M*

Figure 9. Average Precision for the top t tags predicted by RKML using different values of n_s, the number of sampled images used for Nyström approximation. In (c), n_s couldn't be set too large due to the dataset size.

the space of keywords using hierarchical clustering algorithms, (4) Clustering the space of keywords together with the visual features, and (5) Generating binary constraints based on the number of common keywords, *i.e.*, images sharing more than 4 keywords are considered as similar and images sharing no keywords are considered as dissimilar. Note the last one is applicable in LMNN, but not applicable in many other DML algorithms. For example, RCA [1] and DCA [3] divide image set into groups where images within a group are considered as similar and images from different groups are considered as dissimilar; but this method is not able to generate such groups. We observe that these methods yield essentially the same performance reported in our study, as shown in Table 7.

AP@t(%)	t=1	t=4	t=7	t=10
Method 1	20.7 ± 0.2	15.3 ± 0.2	12.4 ± 0.12	10.6 ± 0.10
Method 2	20.6 ± 0.3	15.2 ± 0.2	12.4 ± 0.11	10.6 ± 0.09
Method 3	20.8 ± 0.2	15.4 ± 0.1	12.5 ± 0.05	10.7 ± 0.04
Method 4	19.7 ± 0.2	14.6 ± 0.1	11.9 ± 0.06	10.2 ± 0.06
Method 5	21.3 ± 0.4	15.9 ± 0.3	12.8 ± 0.20	11.0 ± 0.14

Table 7. Comparison of different methods of generating binary constraints that are applied in baseline DML algorithm LMNN for the top t annotated tags on the Flickr1M dataset. Method 1 clusters the space of keywords, method 2 considers the class assignments as binary constraints, method 3 clusters the space of keywords using hierarchical clustering algorithms, method 4 clusters the space of keywords together with the visual features, and method 5 considers images sharing more than 4 keywords as similar and images sharing no keyword as dissimilar.

References

[1] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning a mahalanobis metric from equivalence constraints. *JMLR*, 6:937–965, 2005.

[2] M. W. Berry and M. Browne. *Understanding search engines: mathematical modeling and text retrieval*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1999.

[3] S. Hoi, W. Liu, M. Lyu, and W. Ma. Learning distance metrics with contextual constraints for image retrieval. In *CVPR*, 2006.

[4] C. A. Micchelli, Y. Xu, and H. Zhang. Universal kernels. *JMLR*, 6:2651–2667, 2006.

[5] B. Schölkopf and A. J. Smola. *Learning with kernels: support vector machines, regularization, optimization and beyond*. MIT Press, 2002.

[6] S. Smale and D.-X. Zhou. Geometry on probability spaces. *Constructive Approximation*, 30(3):311–323, 2009.

[7] G. W. Stewart and J.-G. Sun. Matrix Perturbation Theory (Computer Science and Scientific Computing). 1990.

[8] K. Weinberger, J. Blitzer, and L. Saul. Distance metric learning for large margin nearest neighbor classification. In *NIPS*, 2006.