Multi-task Learning via Non-sparse Multiple Kernel Learning

Wojciech Samek[1 2⋆] and Alexander Binder[1] Motoaki Kawanabe[2 1 ⋆⋆]

[1] Technical University of Berlin, Franklinstr. 28 / 29, 10587 Berlin, Germany,
wojciech.samek@tu-berlin.de alexander.binder@tu-berlin.de,
[2] Fraunhofer Institute FIRST, Kekuléstr. 7, 12489 Berlin, Germany,
motoaki.kawanabe@first.fraunhofer.de

Abstract. In object classification tasks from digital photographs, multiple categories are considered for annotation. Some of these visual concepts may have semantic relations and can appear simultaneously in images. Although taxonomical relations and co-occurrence structures between object categories have been studied, it is not easy to use such information to enhance performance of object classification. In this paper, we propose a novel multi-task learning procedure which extracts useful information from the classifiers for the other categories. Our approach is based on non-sparse multiple kernel learning (MKL) which has been successfully applied to adaptive feature selection for image classification. Experimental results on PASCAL VOC 2009 data show the potential of our method.

Keywords: Image Annotation, Multi-Task Learning, Multiple Kernel Learning

1 Introduction

Recognizing objects in images is one of the most challenging problems in computer vision. Although much progress has been made during the last decades, performance of state-of-the art systems are far from the ability of humans. One possible reason is that humans do incorporate co-occurrences and semantic relations between object categories into their recognition process. On the contrary, standard procedures for image categorization learn one-vs-rest classifiers for each object class independently [2].

In this paper, we propose a two-step *multi-task learning (MTL)* procedure which can find out useful information from the classifiers for the other categories based on *multiple kernel learning (MKL)* [6], and its non-sparse extension [4]. In the first step we train and apply the classifiers independently for each class and construct extra kernels (similarities between images) from the outputs. In the second step we incorporate information from other categories by applying MKL with the extended set of kernels. Our approach has several advantages over standard MTL methods like Evgeniou *et al.* [3],

⋆ né Wojcikiewicz

⋆⋆ We thank Klaus-Robert Müller for valuable suggestions. This work was supported by the Federal Ministry of Economics and Technology of Germany under the project THESEUS (FKZ 01MQ07018) and by the FP7-ICT program of the European Community, under the PASCAL2 Network of Excellence (ICT-216886).

namely (1) it does not rely on a priori given similarities between tasks, but learns them via MKL, (2) it uses asymmetric task relations thus avoids negative transfer effects, i.e. good classifiers are not deteriorated by other bad classifiers, which may occur in MTL with symmetric task relations and (3) in contrast to other MTL methods it is scalable. Our experimental results on PASCAL VOC 2009 images show that information from the other classes can improve the classification performance significantly.

The rest of this paper is organized as follows. Section 2 describes related work. In Section 3 we explain MKL and our multi-task learning procedure. Experimental results are described in Section 4 and Section 5 concludes this work and discuss future issues.

2 Related Work

The main goal of multi-task learning is to achieve better performance by learning multiple tasks simultaneously. In general multi-task learning methods can either utilize common structure or use explicit relations between tasks. Methods utilizing common structure in the data can be used for combining multiple features or learning from unlabeled data in a multi-task framework [1, 12]. Using relations between tasks became very popular in last years. For example Evgeniou *et al.* [3] proposed a framework in which relations between tasks are represented by a kernel matrix and multi-task learning is performed by using a tensor product of the feature and task kernel as input for SVM. Similar work can be also found in [9, 11] or in [8] where the authors used Gaussian Processes for learning multiple-tasks simultaneously. All these approaches are theoretically attractive, but have drawbacks which reduce their applicability in practice. The dimensionality of the kernel matrix increases (as square) with the number of tasks, thus these methods are intractable for many real-world problems. Further, it is necessary to determine task similarities appropriately in advance and in contrast to our method these approaches assume a symmetric relationship between the tasks, but in practice a gain from task A on task B may incur a loss from task B on task A.

Our work is, in philosophy, close to Lampert and Blaschko [5] who applied MKL to multi-class object detection problems. However, their procedure cannot be used for object categorization where detection is not the primal interest and no bounding boxes of objects are available.

3 Multi-Task Learning via MKL

3.1 Multiple Kernel Learning

In image categorization, combining many kernels $K_j(\mathbf{x}, \bar{\mathbf{x}})$, (similarity measures between images $\mathbf{x}$ and $\bar{\mathbf{x}}$) for $j = 1, \ldots, m$ constructed from various image descriptors has become a standard procedure. Multiple kernel learning (MKL) is a method which can choose the optimal weights $\{\beta_j\}_{j=1}^m$ of the combined kernel $\sum_{j=1}^m \beta_j K_j(\mathbf{x}, \bar{\mathbf{x}})$ and learn the parameters of support vector machine (SVM) simultaneously (see [6]).

Originally MKL imposes a 1-norm constraint $\|\beta\|_1 = \sum_j \beta_j = 1$ on the mixing coefficients to enforce *sparse* solutions. Recently, Kloft et al. [4] extended MKL to allow *non-sparse* mixing coefficients by employing a generalized p-norm constraint $\|\beta\|_p = \left(\sum_j \beta_j^p\right)^{1/p} \leq 1$ and showed that it outperforms the original one in practice.

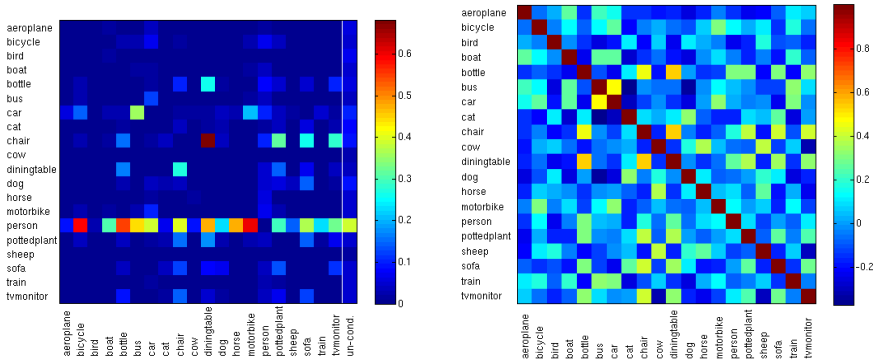

Fig. 1. Left Panel: Co-occurrence relations between the 20 categories of VOC 2009. The entries are $P(\text{class\_row}|\text{class\_column})$ except for the last column with $P(\text{class\_row})$. If a conditional probability is higher than its unconditioned value, then class_row appears frequently with class_column together (e.g. diningtable and chair). In the opposite case both categories are rather exclusive to each other (e.g. aeroplane and person). Right Panel: Kendall rank correlation scores.

3.2 Multi-Task Learning

In a multi-task learning problem we obtain samples $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$ where the vector $\mathbf{y}$ consists of the binary labels $y_c \in \{+1, -1\}$ of C different categories (e.g. visual concepts). Since some of these visual concepts may have semantic relations and can appear simultaneously in images, it is natural to incorporate such relations into learning processes. In order to see the pair-wise co-occurrence between the 20 object categories, we plotted in Figure 1 (left panel) the conditional probabilities of the classes in the rows given those in the columns, where the diagonal entries with probabilities 1 are excluded for better visibility. We see that for instance diningtable and chair appear together frequently, while classes such as cat and cow are rather exclusive to each other.

Apart from co-occurrence there exist other factors characterizing class-relationships. The right panel of Figure 1 shows Kendall rank correlation score τ

$$\tau = \frac{(\#\text{concordant pairs}) - (\#\text{disconcordant pairs})}{\frac{1}{2}n(n-1)},$$

where n is the number of samples and a pair of samples is concordant, if the orders of the two classification scores agree. For instance, although aeroplane and boat appear together very rarely, their classification scores are positively correlated, because they often share similar backgrounds (e.g. blue sky). Multi-task procedures aim at improving classification performances by uncovering statistical relations overlooked by class-wise binary solutions. However, in competitions on object categorization like PASCAL VOC 2009, there have been almost no submissions using the additional label interactions to improve performance over one-vs-rest classifiers.

We tackle this by constructing a classifier which incorporates information from the other categories via MKL for each class separately. Our procedure consists of two steps.

First Stage: For each binary problem we compute the SVM outputs using the average feature kernel.

Second Stage: (1) For each class we construct an output kernel based on the SVM scores from first stage. (2) For each class, we apply sparse or non-sparse MKL with the feature kernels and the output kernels from the other categories.

We measure similarities between the SVM outputs by using the exponential kernel

$$\widetilde{K}_c(\mathbf{x}_i, \mathbf{x}_j) = \exp\left[-\left\{s_c(\mathbf{x}_i) - s_c(\mathbf{x}_j)\right\}^2\right], \tag{1}$$

where $s_c(\mathbf{x}_i)$ is the score of SVM for the c-th category for the i-th example $\mathbf{x}_i$. We neither normalized the scores, nor optimized kernel width further. It must be noted that we cannot compare SVM outputs of training examples with those of test samples, because their statistical properties are not the same. In particular, most of the training images become support vectors whose SVM outputs almost coincide with their labels. In this paper, we deployed 5-fold cross-validation to obtain reasonable scores for the training images, while for the validation and test images, one SVM with the entire training data was used to compute the classifier outputs.

4 Experimental Results

4.1 Experimental Setup

We used the data set of PASCAL VOC 2009 [2] which consists of 13704 images and 20 object classes with an official split into 3473 training, 3581 validation, and 6650 test examples. We deployed the bag-of-words (BoW) image representations over various color channels and the spatial pyramid approach [7] using SIFT descriptors calculated on different color channels (see [10]). We used a vocabulary of size 4000 learned by k-means clustering and applied a χ^2 kernel which was normalized to unit variance in feature space. Average precision (AP) was used as performance measure.

We created 10 random splits of the unified training and validation sets into new smaller data sets containing 1764 training, 1763 validation, and 3527 test images. Since the true labels of the original test images have not been disclosed yet, we excluded these from our own splits. We remark that the AP scores reported in this paper are not comparable with those by the VOC 2009 winners, because the training sample size here is much smaller than that of the official split. For each of the 10 splits, the training images were used for learning classifiers, while with its validation part we selected the SVM regularization parameter C based on the AP score. The regularization constant C is optimized class-wise from the candidates $\{2^{-4}, 2^{-3}, \ldots, 2^4\}$.

The following three options were tested in the second step of our method: 1-norm MKL, 2-norm MKL and average kernel SVM with 15 BoW and 19 output kernels.

4.2 Performance comparison

In the first experiment we compare the performance of the three multi-task learning strategies (denoted with '`M`') using both the 15 feature and the 19 output kernels with

Split	B_ave	B_2mkl	B_1mkl	M_ave	M_2mkl	M_1mkl
1	0.4949	0.4932	0.4726	0.4958	**0.5033**	0.4737
2	0.4845	0.4845	0.4683	0.4818	**0.4926**	0.4675
3	0.4893	0.4882	0.4775	0.4879	**0.4945**	0.4756
4	0.4963	0.4957	0.4804	0.4938	**0.5004**	0.4804
5	0.4862	0.4910	0.4704	0.4910	**0.5000**	0.4781
6	0.4908	0.4896	0.4783	0.4929	**0.5029**	0.4779
7	0.4875	0.4905	0.4665	0.4962	**0.5012**	0.4685
8	0.4866	0.4875	0.4736	0.4857	**0.4970**	0.4753
9	0.4937	0.4959	0.4801	0.4980	**0.5067**	0.4852
10	0.4994	0.4983	0.4768	0.4887	**0.5030**	0.4788
average	0.4909	0.4914	0.4745	0.4912	**0.5002**	0.4761

Table 1. Average AP results for 10 splits. The combination of feature and output kernels with non-sparse MKL outperforms all other settings. The performance gains of M_2mkl are all significant.

three different baselines (denoted with 'B') with kernels computed only from BoW features. The mean AP scores for the 10 splits are summarized in Table 1. Note that a comparison with standard MTL methods like [3] was not possible as it is computationally infeasible.

Two observations can be made from the results. First of all we see that the multi-task learning method with 2-norm MKL (M_2mkl) outperforms the other settings in all n = 10 runs. An application of the t-test shows that the performance gains are all highly significant e.g. the difference ΔX between M_2mkl and the average feature kernel baseline B_ave is significant with p-value less than 0.1% as

$$t = \sqrt{n}\frac{E[\Delta X]}{\sqrt{Var[\Delta X]}} = 7.4247$$

is larger than $t(1-\frac{\alpha}{2}, n-1)$ = 6.86.

The second observation which can be made from the results is that performance decreases when using 1-norm MKL. Note that this effect occurs among the 'B' options as well which do not use any kernels from SVM outputs. This result is consistent with [4] who showed that the non-sparse MKL often outperforms 1-norm MKL in practice. Especially in difficult problems like object classification sparsity is often not the best choice as it ignores much information.

An interesting fact is that the performance gain is not uniformly distributed over the classes. The left panel of Figure 2 shows the average relative performance change between M_2mkl and B_ave over all 20 VOC classes. The largest average gain can be observed for the classes sheep, dog, diningtable, horse, motorbike and cow. Using the t-test we can show that the performance gain for the classes aeroplane, bicycle, bird, boat, car, cat, diningtable, dog, motorbike, person, sheep and train are significant with α = 5% and the gain for classes bird, boat, dog, motorbike, person and train is even significant with p-value less than 1%.

So the question now is *can we identify the classes (output kernels) which are responsible for the performance gain of these classes ?*

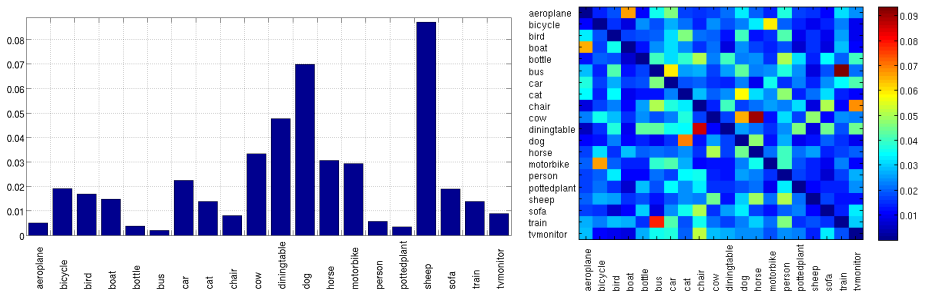

Fig. 2. Left Panel: Relative performance change (average over 10 runs) per class between the 2-norm multi-task MKL and average kernel baseline. Right Panel: Average MKL weights β. The classes in the row indicate the classification tasks, while those in the column are the output kernels. The strength of contribution is not symmetric e.g. see chair - diningtable.

4.3 Interactions between object categories

The kernel weights of MKL give a hint to what extent a classifier uses the output information from another classifier. The right panel of Figure 2 shows the average weights of MKL and we see some prominent links, e.g. train $\rightarrow$ bus, horse $\rightarrow$ cow, chair $\rightarrow$ diningtable. In order to visualize the relations, we created a class-relation graph with the 15 largest kernel weights β. This graph is asymmetric i.e. the output of the class A classifier may be important for classification of class B (arrow from A to B), but not vice versa. It is interesting that this graph although created from MKL kernel weights reveals a semantically meaningful grouping of the classes into: **Animals** (horse, cow, dog, cat), **Transportation** (bus, car, train), **Bikes** (bicycle, motorbike), **Home** (chair, tvmonitor, sofa, diningtable), **Big bluish areas** (aeroplane, boat).

We can think of at least two reasons why the classifier output can help classifiers in the same group. First, co-occurrence relationships can provide valuable information e.g. a chair in the image is an evidence for a diningtable. The second reason is that objects from different classes may have similar appearance or share similar context e.g. images with aeroplanes and boats often contain a large bluish area, the sky and water respectively, so that the outputs of aeroplane classifier may help to classify boats.

4.4 Ranking of images

When we compare image rankings of the baseline and our classifiers in more detail, we gain interesting insights, e.g. the cat $\rightarrow$ dog and chair $\rightarrow$ diningtable rankings are analysed in Figure 4. On the horizontal axis we divide the test set into 35 groups of 100 images based on cat (or chair) scores and create a box plot of the rank difference of dog (resp. diningtable) outputs between the two procedures for each group. In both cases, our method provide better ranks (i.e. positive) in some interval from rank 101 (101 - 1300 for dog and 101 - 600 for diningtable). We conjecture that this is caused mainly by similarities between objects (e.g. cat and dog) and/or backgrounds (e.g. indoor scene). On the other hand, the top group (1 - 100) has rather negative shifts (i.e. our method

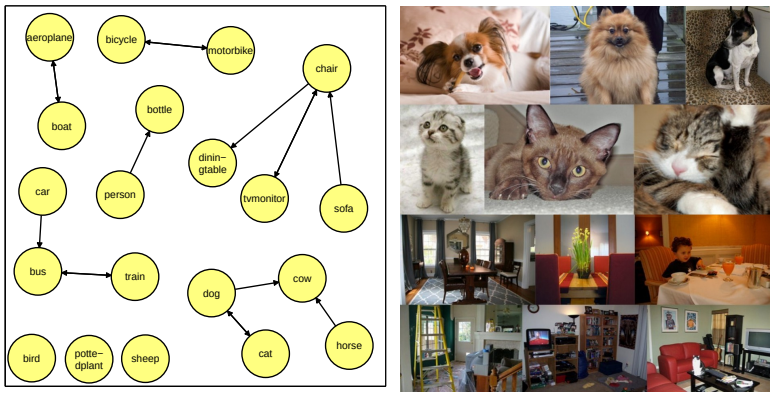

Fig. 3. Left Panel: Class-relation graph computed from MKL weights showing the 15 strongest relations. An arrow from A to B indicates that B is using the output kernel from A with high kernel weight. Right Panel: Images with substantial rank changes. Top images using dog classifier (upper: 297→207, 50→4, 140→60, lower: 33→164, 86→280, 108→1057), Bottom images using diningtable classifier (upper: 28→15, 486→345, 30→6, lower: 36→63, 35→61, 9→36).

gave lower ranks than the baseline) for dog, while shows more positive changes for diningtable. It can be possible that this behavior is caused by co-occurrence relations.

Finally, we show in the right panel of Figure 3 example images which had large differences in rankings by the two methods. The three images in the upper row of each group got higher ranks by our classifier and contain the correct objects, while the other three in the lower row had worse ranks and the object class does not appear. For the images containing the correct objects, the proposed method gave better ranking than the baseline. Among dog (or diningtable) images, 63% (60% resp.) obtained better ranks with median improvement +36 (+14 resp.). On the other hand, we also observed that mutually-exclusive categories may reduce false positives, e.g. among non-diningtable images containing cat, 73% had worse ranks with median difference -139.

5 Conclusions

The multi-task learning approach presented in this paper allows to automatically incorporate relations between object categories into the learning process without a priori given task similarities, by using the output information from other classifiers. It can potentially capture co-occurrence information as well as visual similarities and common backgrounds. We showed that our approach with non-sparse MKL not only significantly outperforms the baselines, but also allows to extract semantically meaningful relations between categories. Furthermore, we analysed the rankings obtained by the proposed method in comparison with those by the non-multi-task baseline. It reveals that interactions between different categories are affected by multiple factors, but can be captured by our method in a non-linear fashion through the output kernels.

In future research we plan to combine our idea with a boosting method and compare it with existing approaches which model relationships between classes explicitly.

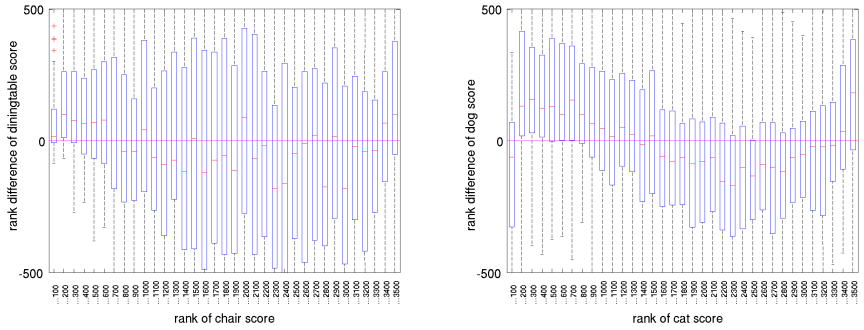

Fig. 4. Differences between the baseline and our method of diningtable (dog) ranks conditioned on chair (cat) ranks. On the horizontal axis each group consists of 100 images, i.e. the first group is from rank 1 to 100 of the chair (cat) classification score. The box plots show that till rank 600 the diningtable score tend to be ranked higher by our method than by the SVM baseline probably due to co-occurrence and common indoor context. In the top group of cat, we see a downside shift of the dog score probably because of negative co-occurrence relation, while images with rank 101-1300 of the cat score are ranked higher due to similarities between dog and cat images.

References

1. Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Multi-task feature learning. In *Neural Inf. Proc. Sys.* MIT Press, 2007.
2. M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2009 (VOC2009) Results.
3. T. Evgeniou, C. A. Micchelli, and M. Pontil. Learning multiple tasks with kernel methods. *J. of Mach. Learn. Res.*, 6:615–637, 2005.
4. M. Kloft, U. Brefeld, S. Sonnenburg, P. Laskov, K.-R. Müller, and A. Zien. Efficient and accurate ℓ^p-norm mkl. In *Neural Inf. Proc. Sys.*, 2010.
5. C. Lampert and M. Blaschko. A multiple kernel learning approach to joint multi-class object detection. In *DAGM*, pages 31–40, 2008.
6. G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. El Ghaoui, and M. I. Jordan. Learning the kernel matrix with semidefinite programming. *J. of Mach. Learn. Res.*, pages 27–72, 2004.
7. S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Conf. on Comp. Vision & Pat. Rec.*, 2006.
8. K. Ming, A. Chai, C. K. I. Williams, S. Klanke, and S. Vijayakumar. Multi-task gaussian process learning of robot inverse dynamics. In *Neural Inf. Proc. Sys.*, 2008.
9. D. Sheldon. Graphical multi-task learning, 2008. Neural Inf. Proc. Sys. workshop on strictured input - structured output.
10. K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Trans. Pat. Anal. & Mach. Intel.*, 2010.
11. Wei Wu, Hang Li, Yunhua Hu, and Rong Jin. Multi-task learning: Multiple kernels for multiple tasks. Technical report, Microsoft Research, 2010.
12. Xiao-Tong Yuan and Shuicheng Yan. Visual classification with multi-task joint sparse representation. In *IEEE Conf. on Comp. Vision & Pat. Rec.*, pages 3493 –3500, 2010.